LARVA: Large-scale Analysis of Recurrent Variants in Annotations

Authors: Lucas Lochovsky, Ekta Khurana, Arif Harmanci, and Mark Gerstein

Abstract

We present LARVA (Large-scale Analysis of Recurrent Variants in Annotations), a computational framework for aggregating rare somatic and germline variants from multiple samples on genomic elements. These *recurrent mutations* serve as a measure of an element's mutational burden, and high-burden elements may correspond to important sites of disruptions for diseases like cancer, and therefore may be crucial for understanding diseases' mechanisms and treatment. LARVA enables the discovery of both recurrent somatic and germline variants in the same annotation, which could implicate previously unknown disease-causing variants. In this paper, we explain the concepts of LARVA's framework, and how it functions to identify recurrent mutations and recurrently mutated genome annotations. We illustrate how LARVA may be used to study recurrent mutation patterns in both coding regions and noncoding regulatory elements, and sets of pathways and interaction networks. For the purposes of determining if observed recurrent variation is statistically significant, we introduce a Statistical Assessment Module to assess the statistical significance relative to recurrent variation expected under neutral mutation processes. Starting with an existing exome model of factors that influence the neutral mutation rate, we have developed our model to simulate expected variation across the entire human genome. Our model makes use of whole genome mutation rate influencers such as DNA replication timing, histone marks, whole genome RNA-seq signals, and SNV density. Our system also provides an Analysis Integration Module for the integration of multiple LARVA analyses, for deeper understanding of disease variation. We have applied LARVA's methods to sets of prostate cancer WGS data to demonstrate its usefulness.

Introduction

Genomes of numerous cancer patients have been sequenced (Barbieri 2012, Baca 2013, Grasso 2012), opening up opportunities to identify the underlying genetic causes for cancer phenotypes and develop more effective therapies targeted at specific molecular subtypes of cancer. Most of these studies have been so far focused on identifying mutations and defects in the protein-coding regions, or exomes, of cancer genomes (Baca 2013). However, this approach ignores investigation of potential variation in important noncoding features of the genome.

There are many noncoding genome regions that influence gene transcription. Such features include pseudogenes, some of which are transcribed and can be incorporated into functional transcripts (Pei 2012). There are also various classes of noncoding RNA, such as microRNA (miRNA), small interfering RNA (siRNA), small nuclear RNA (snRNA), and small nucleolar RNA (snoRNA) that bind and regulate transcripts (Esteller 2011). Furthermore, the binding sites of transcription factors,

which are important for regulation of gene expression, can also be affected by somatic mutations in cancer. Finally, various protein factors can bind to stretches of genomic DNA called enhancers that promote gene transcription.

Some computational systems, such as HaploReg (Ward 2011) and RegulomeDB (Boyle 2012), were previously developed to determine the effect of GWAS variants on noncoding annotations. HaploReg intersects the variants of WGS samples with a fixed series of noncoding regulatory elements in the human genome, determines the variants' effects on noncoding regulatory motifs, and indicates the chromatin state of the genomic region to which each variant maps. RegulomeDB further develops this idea by expanding the range of genome annotations used to include experimentally verified regulatory regions, ChIP-seq-derived transcription factor (TF) binding sites, eQTL, and DNase footprinting.

Also important for understanding the effects of disruptive cancer mutations is the placement of cancer-mutated genes in their systems-level contexts. Identifying the pathways and interactions in which the products of mutated genes participate is often crucial to seeing precisely how cellular functions are being disrupted by cancer (Vandin 2011). Protein interaction networks have also proven useful for characterizing cancer disruption: disrupted subnetworks of interacting proteins have been used to more accurately classify subtypes of breast cancer in Chuang *et al.* (2007).

Recent computational systems that focus on the cancer pathway disruption include cBio (Cerami 2012) and Multi-Dendrix (Leiserson 2013). cBio starts with variant datasets, and a database of genes and their pathway membership information. The cBio system then identifies those pathways mutated with high coverage and high mutual exclusivity. High coverage refers to the presence of mutations in a large proportion of samples, and high exclusivity means that many of the highly damaging, driver mutations appear in mutually exclusive samples, owing to the sufficiency of mutating just one part of a pathway to nullify its function. Multi-Dendrix extends these ideas by introducing new algorithms to find arbitrary sets of genes that exhibit high coverage and mutual exclusivity of variants, rather than being limited to previously established pathways. GEMINI (Paila 2013) is another general system that manages variant call sets and genome annotation sets through an SQL database, and allows users to formulate their own SQL-based queries over the stored data, allowing a wide range of flexibility for exploring variant data.

Here, we present a computational system that supports the study of cohorts of whole genome sequenced (WGS) disease patient samples. The primary function of LARVA (Large-scale Analysis of Recurrent Variants in Annotations) is to identify recurrent patterns of disease mutations in various genome annotations using WGS data from multiple disease patients, and compute the statistical significance of these findings. Our framework makes use of a relational database system approach to organize, maintain, and operate on genome variant and annotation data in a systematic way. LARVA allows users to investigate recurrent variation patterns that

a set of disease variant data presents in a set of genome annotation data by casting the relevant questions as SQL queries. This framework accommodates a wide range of query types, spanning any genetic disease, and any set of genome annotations one wishes to study.

On a simple level, a mutation recurrence would be a variant at exactly the same position in two individuals. However, this is exceedingly unlikely for rare variants (Durbin 2010). Thus, we will consider mutational burden spread over elements. These elements can be single annotations, such as exons, pseudogenes, noncoding RNA, and regulatory features like promoters and enhancers. On a more complex level, we will consider groups of genes related through a common pathway, or through a protein interaction subnetwork, as a single element, where variants from multiple patients that map anywhere in the gene group represent a recurrence.

LARVA enables the discovery of annotations that contain recurrent somatic variants and recurrent rare germline variants. Elements with both types of mutation recurrence could indicate a functional connection between the overlapping somatic and germline variants. The absence of common variants from these elements would serve as further evidence for a functional connection.

In addition to recurrent variant identification, LARVA offers two additional modules. First, LARVA includes a Statistical Assessment Module, LARVA-SAM, that uses a model of neutral genome evolution to determine the statistical significance of the recurrent mutation patterns that LARVA identifies. Building on a previously developed null model for exomes (Lawrence 2013), we introduce a null model for whole genomes. Secondly, LARVA's Analysis Integration Module (LARVA-AIM) enables further exploration of a LARVA systems-level analysis. When LARVA is used to study recurrent variation in pathways and networks, LARVA-AIM may be employed to place recurrently mutated genes in their pathway and network context. Recurrent gene and pathway/network data are combined to allow one to observe the number of pathways in which a recurrently mutated gene participates, or the number of network neighbors it has.

We have applied LARVA to cancer data to elucidate patterns of recurrent prostate cancer mutations in important noncoding regulatory features of the genome. LARVA has also been used to explore recurrent mutations on a pathway and interaction network level in this data. The following sections describe LARVA's concepts, and their applications to the study of genetic disease.

LARVA Concepts

One of LARVA's important design features is its use of a relational database to manage its data and express recurrent variant exploration as database queries. This arrangement provides users with an efficient, expressive means of organizing LARVA's results and pursuing followup analyses. We have implemented LARVA's relational database features using SQLite.

The core module provides analysis of disease variant calls for patterns of recurrent variation in genome annotations. We shall call this module LARVA-Core. This module has two primary inputs: *variant files* and *annotation files*.

The *variant files*, or *vfiles*, are derived from patients whose genomes have been sequenced, and for which *single nucleotide variants* (SNVs) have been called by comparing the patients' genomes to a reference genome. Each file corresponds to a single patient's variant calls.

The *annotation files*, or *afiles*, are derived from a number of genome annotation sources. *Afiles* we have collected for LARVA analysis include protein-coding exon, pseudogene, and noncoding RNA data from the GENCODE project (Harrow 2012). We also studied transcription factor binding sites derived from a number of sources (Rozowsky 2009, Kheradpour 2012). Finally, we sought to understand cancer variation on a system-wide level by studying recurrent variation in metabolic pathways and protein interaction networks (Kanehisa 2000, Kanehisa 2011, Prasad 2009).

Measures of Mutation

LARVA-Core intersects a set of *vfiles* with a set of *afiles* and identifies two types of recurrent mutations. These include:

- *Recurrent variants:* Overlapping SNVs from multiple samples that fall into at least one *afile* annotation (Fig. 1a). Such mutations may correspond to a critical component of the annotation's function that is important for tumor suppression. These mutations may also be used to classify the subtype and severity of cancer patients (Vandin 2011).
- *Recurrently mutated annotations:* Annotations that contain SNVs from multiple samples that do not necessarily overlap (Fig. 1b). Such annotations may be functionally disruptable in multiple places, and therefore, multiple patients with the same functional disruption may carry SNVs in different places of the same gene.

LARVA-Core's findings are presented using three "Measures of Mutation". These are computed for each annotation, and each *afile* annotation set. They are:

- *Number of samples mutated*: The number of samples represented by SNVs that fall anywhere in the given annotation, or *afile* annotation set.
- *Number of annotations recurrently mutated*: The number of annotations in an *afile* annotation set that are mutated in at least two samples. This is not applicable to individual annotations.

- *Number of recurrent variants*: The number of SNVs from multiple samples that overlap exactly, and fall anywhere in the given annotation, or *afile* annotation set.

LARVA Statistical Assessment Module (LARVA-SAM)

It is important to determine whether the recurrently mutated annotations and recurrent variants of LARVA-Core are statistically significant, in that these patterns are not the result of random, neutral mutation processes. To that end, LARVA has a module for randomly generating sets of cancer variants similar to the actual datasets, and running LARVA-Core on these random datasets to gather information on recurrently mutated annotations and variants that would occur by chance. Hence, a random distribution of the "Measures of Mutation" is generated, and compared to the actual, observed "Measures of Mutation" to determine whether the mutation patterns of the actual datasets are statistically significant.

Random variant generation for whole genome datasets

When LARVA-SAM is used on whole genome variant datasets, LARVA's whole genome neutral mutation "null model" is used, which simulates the distribution of variants expected over a neutrally evolving genome. Our null model is defined as a weight function that assigns weight to discrete partitions of the genome. The factors used in our whole genome model include:

- *DNA replication time:* Early in the DNA replication process, there are more free nucleotides available for DNA repair. As the process continues, this nucleotide pool is depleted, and portions of the genome that are replicated at a late phase are more likely to pick up mutations (Chen 2010).
- *H3K4me1 and H3K4me3 marks:* Schuster-Böckler and Lehner (2012) demonstrated that H3K4me1/me3 marks are anti-correlated with SNV density.
- *Expression level:* More highly expressed genome regions have higher levels of transcription-coupled repair (Barretina 2012).
- *SNV density:* The 1000 Genomes Project has researched differences in mutation rate due to natural population variation. (Durbin 2010).
- *GC bias:* Genome regions with more G and C bases have higher substitution rates (Smith 2002). Incorporation of this factor into our model is under way.

The weight of a region *r* in the genome is defined with the following function:

$$
\begin{aligned}
&weight(r) \\
&\quad = w_1 log\left(CDF\left(reptiming(r)\right)\right) + w_2 log\left(1 - CDF\left(H3K4me1(r)\right)\right) \\
&\quad + w_3 log\left(1 - CDF\left(H3K4me3(r)\right)\right) + w_4 log\left(1 - CDF\left(expression(r)\right)\right) \\
&\quad + w_5 log\left(CDF\left(SNV\_density(r)\right)\right)
\end{aligned}
$$

where

- *r* is a 100,000-bp-long block of the human genome (hg19 build).
- *reptiming*(*r*) is the replication timing of region *r*, according to Chen *et al.* (2010)
- *H3K4me1*(*r*) is the level of histone H3K4 mono-methylation of region *r*, according to ENCODE GM12878 Peak-seq experiments (Dunham 2012).
- *H3K4me3*(*r*) is the level of histone H3K4 tri-methylation of region *r*, according to ENCODE GM12878 Peak-seq experiments (Dunham 2012).
- *expression*(*r*) is the expression level of region *r*, according to the ENCODE's GM12878 RNA-seq track (Dunham 2012).
- *SNV_density*(*r*) is the number of SNVs in region *r*, according to the 1000 Genomes Project (Durbin 2010).
- $w_1 \ldots w_5$ are the weights assigned to each variable to represent differing contribution levels. These can be adjusted to fit the model to the observed contributions of each factor.

CDF here refers to the cumulative distribution function of the expression values, replication timing values, etc. It functions as a percentile ranking of each variable within its respective distribution, and influences *weight*(*r*) accordingly. For example, genes with higher H3K4me1 marks are less likely to be mutated, therefore lower H3K4me1 values will map to higher weights, and higher H3K4me1 values will map to lower weights.

When a region is chosen, the exact variant position is determined by randomly choosing a position in the selected region with uniform probability. This whole genome method of random variant placement represents an extension of Lawrence *et al.'s* (2013) methods to account for the systemic biases in effect on the human exome's neutral mutation rate.

LARVA-SAM will generate a user-specified number, *nrand*, of replicates of the *vfiles* dataset that represents the actual data. Each of these replicate datasets contain the same number of *vfiles* and variants as the original dataset, but have randomized variant positions.

Random variant generation for exome datasets

When LARVA-SAM is used on exome variant datasets, the random variant datasets are derived by simulating the distribution of variants expected for a neutrally evolving exome. Our neutral mutation "null model" is defined as a weight distribution over all genes, where the weight is based on a number of factors that influence their neutral mutation rate (Lawrence 2013). These factors include:

- *Expression level:* As in the whole genome model, more highly expressed genes have higher levels of transcription-coupled repair (Barretina 2012).

- *DNA replication time:* As in the whole genome model, later replicating portions of the genome are more likely to pick up mutations (Chen 2010).
- *Chromatin state:* Genome regions with open chromatin are less likely to be mutated than regions with closed chromatin, likely due to differences in accessibility to DNA repair complexes (Schuster-Böckler 2012).
- *Length:* Longer genes will pick up more variants by chance than shorter genes.

These factors are used to produce a weight for a gene g using the following function:

$$weight(g) = w_1 log\left(1 - CDF\left(expression(g)\right)\right) + w_2 log\left(CDF\left(reptiming(g)\right)\right) + w_3 log\left(1 - CDF\left(chromatin\_state(g)\right)\right) + w_4 log\left(CDF\left(length(g)\right)\right)$$

where

- $expression(g)$ is the expression level of gene g, according to the Cancer Cell Line Encyclopedia's (CCLE) RNA-Seq data (Barretina 2012). This is an average of the expression across all CCLE cancers.
- $reptiming(g)$ is the replication timing of gene g, according to Chen *et al.* (2010).
- $chromatin\_state(g)$ is a measure of how open or closed the chromatin is at gene g, according to Lieberman-Aiden *et al.* (2009).
- $length(g)$ is the length of gene g.
- $w_1 \ldots w_4$ are the weights assigned to each variable to represent differing contribution levels.

Once the gene to place the random variant in has been chosen, the gene's exon coordinates are retrieved, and an exact position for the random variant is determined by selecting one at random from the retrieved exons, with uniform probability. This procedure is repeated for each variant to be generated for the given random variant file.

LARVA-Core runs and Normal distribution fitting

After the random variant generation step, LARVA-SAM will have generated *nrand* random variant datasets. These datasets are used as input for LARVA-Core, generating *nrand* datapoints approximating the expected distribution of each Measure of Mutation. These datapoints are fit to a Normal distribution, and compared to the corresponding Measure of Mutation from the actual *vfile* data to produce a *p*-value, for significance testing.

LARVA Analysis Integration Module (LARVA-AIM)

LARVA-Core may be used for numerous types of analyses, the results of which can be integrated for better understanding of disease variation. To this end, we have

developed the LARVA Analysis Integration Module (LARVA-AIM), designed to facilitate the integration of multiple analyses after significance testing.

To assist in the systems-level analysis of disease variant files, LARVA-AIM may be used to integrate a LARVA-Core gene analysis and a LARVA-Core pathway analysis. LARVA-AIM can take a list of recurrently mutated genes and place them in the pathways in which those genes participate. Additionally, LARVA-AIM can be used to understand recurrently mutated genes in the context of their protein products' interactions. The AIM module can bring recurrently mutated gene analysis data and protein interaction network data together, so users can see the number of interaction partners for each recurrently mutated gene. This enables the identification of potential disease-related network hubs.

Example Workflows of applications using LARVA

By plugging a genetic disease cohort's variant calls into LARVA's *vfiles* parameter, and using different settings of LARVA's *afiles* parameter, one may use LARVA to study a cancer cohort's patterns of recurrent variation over many genome annotations of interest. We illustrate this flexibility with the following examples.

1) ***afiles* = noncoding RNA annotations.** With this setting, LARVA can find potential regulatory drivers from a genetic disease cohort in noncoding RNA (ncRNA). ncRNA annotations may be derived from the GENCODE project (Harrow 2012). Recurrent variants corresponding to putative critical point mutations in ncRNA will be identified, as well as any ncRNA mutated in multiple samples.
2) ***afiles* = KEGG pathways.** Here, one may define an *afile* for each pathway in the KEGG database (Kanehisa 2000, 2011), each containing the pathway members. Under this arrangement, one may study a genetic disease cohort's recurrently mutated pathways using LARVA's annotation set "Measures of Mutation". Once pathways worth closer investigation are identified at this higher level of analysis, one may drill down into the annotation "Measures of Mutation" for those pathways to investigate further.
3) ***afiles* = Transcription factor binding peaks.** Using data on the binding sites of transcription factors from ENCODE Peak-seq experiments (Rozowsky 2009), one may use LARVA to identify recurrent mutations that may lead to expression dysregulation in a genetic disease cohort. By defining an *afile* for each transcription factor, each containing that factor's sites, one may identify both factors and sites that should be studied further.

LARVA Applications to Cancer

We have applied LARVA to studying recurrent variants and recurrently mutated annotations in a number of prostate cancer datasets (Berger 2012, Weischenfeldt 2013, Barbieri 2012, Baca 2013). Our findings have produced new insights into

potential noncoding disruptions in these cancers. LARVA's source code is available to download through Github, at <gersteinlab.github.io/LARVA>.

LARVA Implementation and Computational Efficiency

Fig. 2 illustrates the algorithms behind LARVA-Core's computations. LARVA-Core efficiently processes the intersection of the variant list and the annotation list by sorting each list by chromosome and position on the chromosome. A linear walk of the genome is then performed, and along the way, LARVA-Core will find which variants and annotations overlap, and record the recurrent variants and recurrently mutated annotations encountered. This algorithm is much like the one used in the intersectBed component of the BEDTools software package (Quinlan 2010). However, LARVA-Core takes variants and annotations from an arbitrary number of input files, and therefore requires additional data structures to keep track of the source of each variant and annotation.

When LARVA-Core's computations are complete, the results are produced in an SQLite database, which allows the different types of output to be organized into separate tables, and lets users explore specific portions of the output for the most relevant findings. Although users may query this database directly, a Perl script is available that allows users to construct queries using a wizard-like interface from the command line. Users only need to answer a series of questions, and the script will construct the SQL that corresponds to the user's requested information.

Due to the large number of simulated LARVA-Core runs that LARVA-SAM executes, LARVA-SAM is very compute intensive. Therefore, we have developed a parallel version of LARVA-SAM that leverages multi-core CPUs. Users may specify the number of CPU cores on their machines that LARVA should use. LARVA-SAM will automatically split its LARVA-Core runs across the specified number of cores evenly, and process each batch in parallel. This allows the system to run as efficiently as the available hardware allows.

We conducted timing tests on LARVA-SAM to evaluate how its running time scales with different input sizes. The four parameters that influence LARVA-SAM's running time are:

- Number of variants in *vfiles*
- Number of annotations in *afiles*
- Number of random variant datasets to produce (*nrand*)
- Number of CPU cores to use in parallel

To test the performance effects of the first three variables, we developed tests involving two different sets of *vfiles*, two different sets of *afiles*, and two different *nrand* settings. Table 1 summarizes these inputs and their magnitude. Each of the eight possible combinations of these three variable settings were run on LARVA-SAM, the results of which are illustrated in Fig. 3.

Lucas Lochovsky 3/17/14 12:01 PM
Moved (insertion) [1]

Lucas Lochovsky 3/17/14 3:10 PM
Deleted: and Parallelization

Lucas Lochovsky 3/17/14 12:12 PM
Formatted: Font:Not Bold

Lucas Lochovsky 3/17/14 12:12 PM
Formatted: Font:Not Bold

Lucas Lochovsky 3/17/14 12:12 PM
Formatted: Font:Not Bold

Lucas Lochovsky 3/17/14 12:12 PM
Formatted: Font:Not Bold

Lucas Lochovsky 3/17/14 12:12 PM
Formatted: Font:Not Bold

Lucas Lochovsky 3/17/14 3:21 PM
Formatted: Font:Italic

Lucas Lochovsky 3/17/14 3:21 PM
Formatted: Font:Italic

Lucas Lochovsky 3/17/14 3:21 PM
Formatted: Font:Italic

Lucas Lochovsky 3/17/14 3:20 PM
Formatted: No bullets or numbering

Variant count influence can be seen in Fig. 3 by comparing the running time of the prostate samples vs. KEGG run (*nrand* = 120) and the Grade 4 glioma vs. KEGG run (*nrand* = 120). These correspond to columns 1 and 3 in Fig. 3. The two columns are identical in height, indicating that the variant count does not influence LARVA-SAM's running time. Columns 1 and 2 are indicative of the annotation count's influence on running time, as they correspond to two runs where two different annotation sets were used, with all other parameters being equal. Going from the KEGG annotation set to the GENCODE exon annotation set represents a 3.2x increase in annotations, and the difference between columns 1 and 2 is 4.0x. Hence, LARVA-SAM's running time scales close to linearly with the number of annotations used as input. Finally, the influence of *nrand* can be seen by comparing the pairs of columns offset by 4 (i.e. columns 1 and 5, columns 2 and 6, etc.). The two *nrand* settings differ by a factor of 2.5, and the times in these column pairs also differ by a factor of ~2.5. Therefore, LARVA-SAM's running time scales linearly with the *nrand* setting. Taken together, these results indicate that LARVA-SAM's algorithms are efficiently optimized to scale to large problem sizes.

We also investigated the performance gains derived from spreading LARVA-SAM's workload across multiple CPU cores operating in parallel. Fig. 4 graphs the running time of a LARVA-SAM test query over the number of CPU cores assigned to LARVA-SAM. LARVA-SAM allows the user to specify the number of parallel processes to create for LARVA-SAM's computations, which are automatically spread evenly across the CPU cores available. LARVA-SAM's performance gains are consistent with those of many parallel applications: the addition of the first few cores after the first offers tremendous performance gains, but these gains diminish as the overhead costs of maintaining so many processes at once begins to dominate.

Given these diminishing performance gains, we investigated further to determine if there is an optimal number of CPU cores to use. We determined this by taking measurements between pairs of LARVA-SAM runs where the number of CPU cores was varied. The performance gain per CPU core added between two LARVA-SAM runs r_1 and r_2, where r_2 used more CPU cores than r_1, is calculated as follows:

$$\frac{\frac{|r_2.running\_time - r_1.running\_time|}{r_1.running\_time}}{r_2.ncpu - r_1.ncpu}$$

This expresses the percent gain in performance that r_2 has over r_1 for which each additional core is responsible. We used this measure of performance gain in a series of tests where LARVA-SAM was run using the prostate sample collection as the *vfiles* and the GENCODE exons as the *afiles*. Fig. 5 graphs the performance gain per core added at different CPU core counts using *nrand* = 120 (a) and *nrand* = 300 (b). From both graphs it is clear that there is an "elbow": a point at which the performance gain per core added decreases sharply, and the performance benefit per core becomes minimal. However, it is also clear that at higher *nrand* settings, this elbow

Lucas Lochovsky 3/27/14 9:41 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 3:54 PM
Formatted: Font:Italic

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Font:Italic

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Subscript

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Font:Italic

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Subscript

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Font:Italic

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Subscript

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Font:Italic

Lucas Lochovsky 3/27/14 10:10 PM
Formatted: Subscript

Lucas Lochovsky 3/27/14 10:19 PM
Formatted: Font:Italic

Lucas Lochovsky 3/27/14 10:19 PM
Formatted: Subscript

Lucas Lochovsky 3/27/14 10:19 PM
Formatted: Font:Italic

Lucas Lochovsky 3/27/14 10:19 PM
Formatted: Subscript

increases. For an *nrand* in the thousands range, which is recommended for accurately deriving the expected variant distributions of neutrally evolving genomes, we expect that using as many CPU cores as possible will yield significant performance gains.

Choice of *nrand* Parameter Setting

Given the computational expense involved with running LARVA-SAM for a large number of random datasets *nrand*, we investigated the influence of *nrand* on the stability of the *p*-values produced by the LARVA-SAM significance tests. Specifically, we looked at whether the *p*-values produced by LARVA-SAM cross the commonly used significance thresholds of 0.05 and 0.01 as *nrand* increases.

Our analyses used the prostate sample collection described in Table 1 for the *vfiles*, and used the KEGG pathway data as the *afiles*, also described in Table 1. As we varied the *nrand*, we observed how many *p*-values crossed a significance threshold between consecutive runs. This analysis is demonstrated using a subset of our *p*-value data as presented in Table 2. This data was derived from running LARVA-SAM(prostate sample collection, KEGG pathways) using an *nrand* of 500 and 1000. For each pathway, the expected distribution of each Measure of Mutation was computed, and the findings from those distributions have been summarized in this table. *P*-values from the same analysis done at different *nrand* settings were compared to find those that crossed a significance threshold (i.e. *p*=0.05 or 0.01, which are the two most commonly used thresholds). For example, in Table 2, the p-value of the neuroactive ligand receptor interaction pathway at *nrand*=500 is over 0.05, but at *nrand*=1000, it is below 0.05. It is the only analysis in this sample data that represents a "significance threshold crossing". We generated *p*-values running the prostate sample data against KEGG pathways for a range of *nrand* settings to discover the setting at which the *p*-values had stabilized enough so that no significance threshold crossing were observed for higher *nrand* settings.

Fig. 6 graphs the significance threshold crossings at *p*=0.05 (a) and *p*=0.01 (b). Each bar cluster indicates the crossings observed from increasing the *nrand* as indicated in the horizontal axis. At both significance thresholds, there are no more crossings after *nrand* = 2000. Therefore, we conclude that 2000 simulated datasets is sufficient to produce reliable *p*-values in LARVA-SAM's significance testing.

Lucas Lochovsky 3/31/14 3:58 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 3:58 PM
Formatted: Font:Italic

Lucas Lochovsky 3/31/14 4:14 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:46 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:46 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:48 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:49 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 11:18 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 11:18 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 11:25 AM
Formatted: Font:Italic

Lucas Lochovsky 3/17/14 12:04 PM
Deleted: We have shown that this parallel implementation greatly speeds up the running time for a large number of LARVA-Core runs.

Discussion

In this paper, we have introduced a new computational framework for exploring patterns of recurrent mutation across somatic and rare germline variants. LARVA is designed to be used to explore a broad range of genome annotations to uncover the ones that are mutated across many samples, making it possible to predict putative drivers of genetic disease, and prioritize these predicted drivers for more rigorous downstream analysis. This may lead to faster identification of important targets that may be used to suppress disease in therapies and drugs.

Using a relational database design, LARVA is easily adaptable to many different types of analyses. It may be used to study recurrent mutation patterns across genes, pseudogenes, noncoding RNA, and various noncoding regulatory elements. This ability to study noncoding mutation serves as an important supplement to the many exome-focused studies that have been conducted so far on genetic diseases, such as cancer. LARVA may also be used to study genetic diseases at a systems level, with analyses on pathways and interaction networks possible.

Furthermore, we have developed LARVA-SAM, a module designed to compare observed variant file recurrent mutation patterns to a simulated distribution of variants generated from a neutrally evolving genome model. This comparison allows users to identify genome annotations that are mutated in a higher number of samples, or have a higher number of recurrent variants, than expected under neutral evolution, indicating possible cancer involvement. Finally, we have created LARVA-AIM, a module with the purpose of bringing together recurrent mutation data from multiple types of analyses to shed deeper understanding on features with probable connections to cancer disruption processes.

Future Work

In addition to recurrence information, functional annotation is important to assessing a variant's likelihood of disease association (Khurana 2013). In the future, we plan to add functional annotation capabilities to LARVA, enabling the filtering of results for recurrent variants and recurrently mutated annotations more relevant to diseases. We will also continue to improve LARVA's algorithms and LARVA's user interface. As the amount of genetic data increases, it will be important to further optimize LARVA's computational efficiency, and therefore we are investigating these issues for future iterations of LARVA. Also, we will continue to gain insights by applying LARVA to additional cancer types and subtypes. In the long term, we envision LARVA becoming increasingly useful for elucidating important insights and understanding about all types of genetic diseases.

Lucas Lochovsky 3/17/14 12:01 PM

Moved up [1]: LARVA Computational Efficiency and Parallelization Due to the large number of simulated LARVA-Core runs that LARVA-SAM executes, LARVA-SAM is very compute intensive. Therefore, we have developed a parallel version of LARVA-SAM that leverages multi-core CPUs. Users may specify the number of CPU cores on their machines that LARVA should use. LARVA-SAM will automatically split its LARVA-Core runs across the specified number of cores evenly, and process each batch in parallel. This allows the system to run as efficiently as the available hardware allows. We have shown that this parallel implementation greatly speeds up the running time for a large number of LARVA-Core runs.

References

1. Baca, S. C. *et al.* Punctuated Evolution of Prostate Cancer Genomes. *Cell* **153,** 666–677 (2013).
2. Barbieri, C. E. *et al.* Exome sequencing identifies recurrent SPOP, FOXA1 and MED12 mutations in prostate cancer. *Nature Genetics* **44,** 685–689 (2012).
3. Barretina, J. *et al.* The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature* **483,** 603–307 (2012).
4. Berger, M. F. *et al.* The genomic complexity of primary human prostate cancer. *Nature* (2011).

5. Boyle, A. P. *et al.* Annotation of functional variation in personal genomes using RegulomeDB. *Genome Research* **22,** 1790–1797 (2012).
6. Cerami, E. *et al.* The cBio Cancer Genomics Portal: An Open Platform for Exploring Multidimensional Cancer Genomics Data. *Cancer Discovery* **2,** 401–404 (2012).
7. Chen, C. L. *et al.* Impact of replication timing on non-CpG and CpG substitution rates in mammalian genomes. *Genome Research* **20,** 447–457 (2010).
8. Chuang, H.-Y., Lee, E., Liu, Y.-T., Lee, D. & Ideker, T. Network-based classification of breast cancer metastasis. *Mol Syst Biol* **3,** (2007).
9. D'Antonio, M., & Ciccarelli F.D. Integrated analysis of recurrent properties of cancer genes to identify novel drivers. *Genome Biology* **14**:R52 (2013).
10. DePristo, M. A. *et al.* A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics* **43,** 491–498 (2011).
11. Dunham, I. *et al.* An integrated encyclopedia of DNA elements in the human genome. *Nature* **489,** 57–74 (2012).
12. Durbin, R. M. *et al.* A map of human genome variation from population-scale sequencing. *Nature* **467,** 1061–1073 (2010).
13. Esteller, M. Non-coding RNAs in human disease. *Nature Reviews Genetics* **12,** 861–874 (2011).
14. Grasso, C. S. *et al.* The mutational landscape of lethal castration-resistant prostate cancer. *Nature* **487,** 239–243 (2012).
15. Harrow, J. *et al.* GENCODE: The reference human genome annotation for The ENCODE Project. *Genome Research* **22,** 1760–1774 (2012).
16. Kanehisa, M. and Goto, S.; KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.* **28**, 27-30 (2000).
17. Kanehisa, M., Goto, S., Sato, Y., Furumichi, M. & Tanabe, M. KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Research* **40,** D109–D114 (2011).
18. Keshava Prasad, T. S. *et al.* Human protein reference database—2009 update. *Nucleic acids research* **37,** D767 (2009).
19. Kheradpour, P. Computational regulatory genomics: motifs, networks, and dynamics. (2012). at <http://18.7.29.232/handle/1721.1/70871>
20. Khurana, E. *et al.* Integrative Annotation of Variants from 1092 Humans: Application to Cancer Genomics. *Science* **342,** 1235587–1235587 (2013).
21. Krauthammer, M. *et al.* Exome sequencing identifies recurrent somatic RAC1 mutations in melanoma. *Nature Genetics* (2012). doi:10.1038/ng.2359
22. Lawrence, M. S. *et al.* Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature* **499,** 214–218 (2013).
23. Leiserson, M. D. M., Blokh, D., Sharan, R. & Raphael, B. J. Simultaneous Identification of Multiple Driver Pathways in Cancer. *PLoS Computational Biology* **9,** e1003054 (2013).
24. Lieberman-Aiden, E. *et al.* Comprehensive Mapping of Long-Range Interactions Reveals Folding Principles of the Human Genome. *Science* **326,** 289–293 (2009).

25. McKenna, A. *et al.* The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Research* **20,** 1297–1303 (2010).
26. Paila, U., Chapman, B. A., Kirchner, R. & Quinlan, A. R. GEMINI: Integrative Exploration of Genetic Variation and Genome Annotations. *PLoS Computational Biology* **9,** e1003153 (2013).
27. Pei, B. *et al.* The GENCODE pseudogene resource. *Genome Biol* **13,** R51 (2012).
28. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26,** 841–842 (2010).
29. Rozowsky, J. *et al.* PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. *Nat Biotechnol* **27,** 66–75 (2009).
30. SAM/BAM Format Specification Working Group, The. The SAM/BAM Format Specification (v1.4-r985). Sourceforge.net. 9 Sep 2009. Web. 19 July 2013. <http://samtools.sourceforge.net/SAM1.pdf>
31. Schuster-Böckler, B. & Lehner, B. Chromatin organization is a major influence on regional mutation rates in human cancer cells. *Nature* **488,** 504–507 (2012).
32. Smith, N. G. C. Deterministic Mutation Rate Variation in the Human Genome. *Genome Research* **12,** 1350–1356 (2002).
33. Vandin, F., Upfal, E. & Raphael, B. J. De novo discovery of mutated driver pathways in cancer. *Genome Research* (2011). doi:10.1101/gr.120477.111
34. Ward, L. D. & Kellis, M. HaploReg: a resource for exploring chromatin states, conservation, and regulatory motif alterations within sets of genetically linked variants. *Nucleic Acids Research* **40,** D930–D934 (2011).
35. Weischenfeldt, J. *et al.* Integrative Genomic Analyses Reveal an Androgen-Driven Somatic Alteration Landscape in Early-Onset Prostate Cancer. *Cancer Cell* **23,** 159–170 (2013).

Lucas Lochovsky 3/17/14 1:31 PM
Formatted: Normal, No bullets or numbering

Figures

Fig 1a: Recurrent variants are single nucleotide variants (SNVs) from multiple samples that overlap in a single annotation.

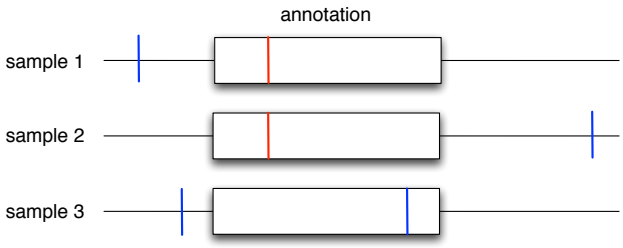


Fig 1b: Recurrently mutated annotations contain variants from multiple samples that are positioned anywhere within the annotation boundaries.

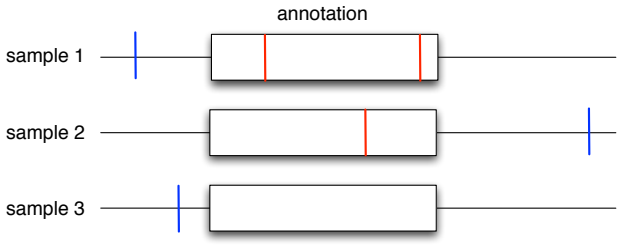


Fig. 2: A schematic of the algorithms behind the LARVA-Core module.

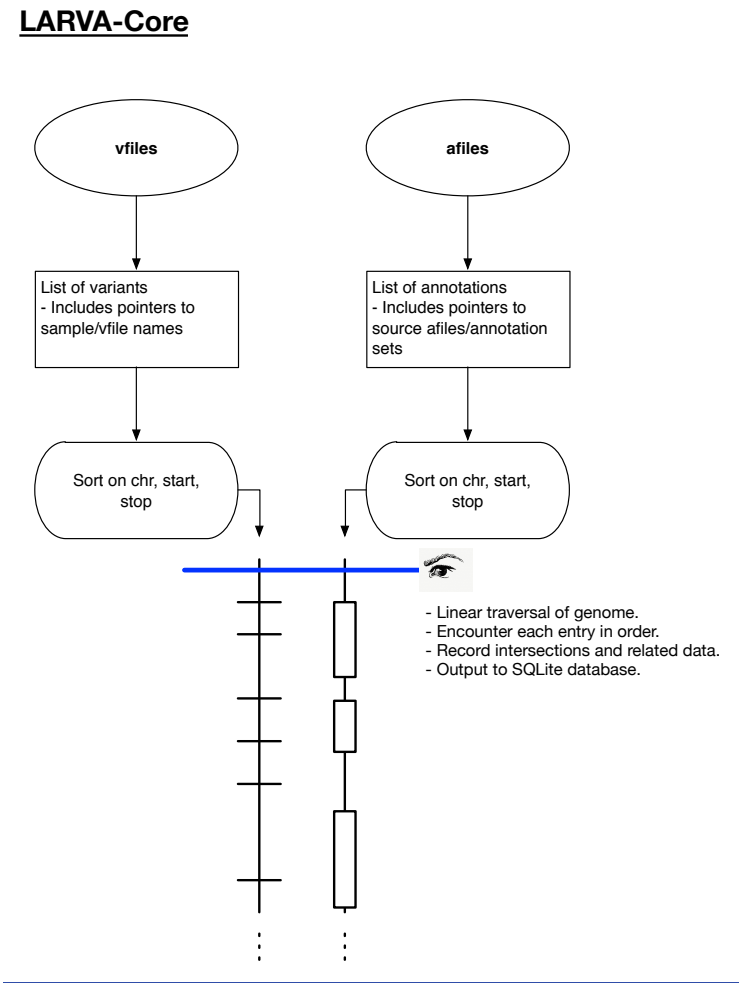


Fig. 3: A series of timing tests for LARVA-SAM, varying the number of input variants, input annotations, and number of random variant datasets to produce to determine how performance scales with these parameters.

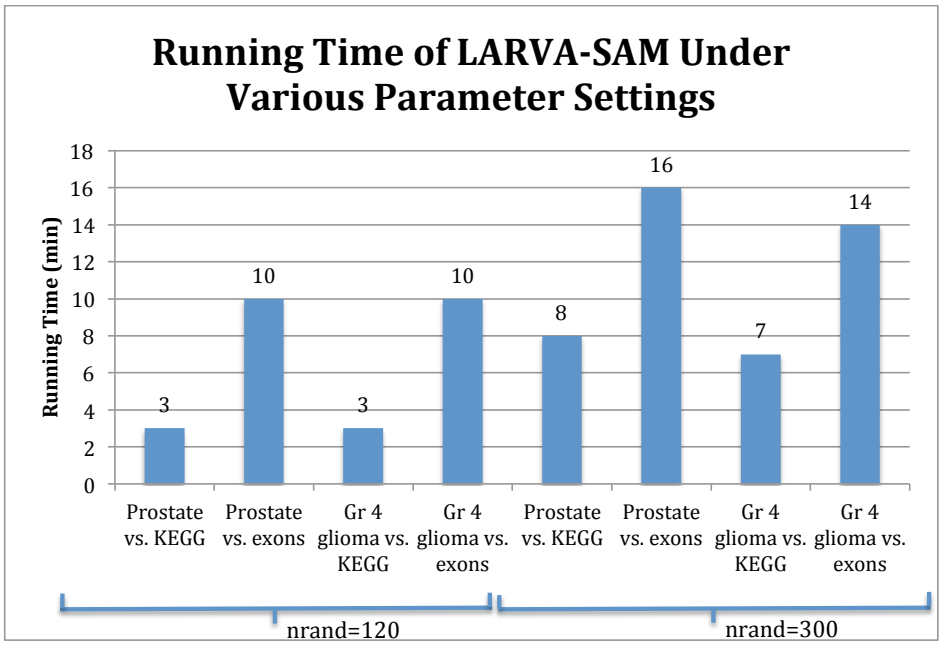


Fig. 4: A graph of the running time with respect to the number of parallel CPU cores used for a LARVA-SAM query of the prostate sample collection (vfiles) against the KEGG pathways (afiles), with a number of random datasets *nrand* of 180. The performance gained relative to the number of CPU cores added steadily decreases, as expected.

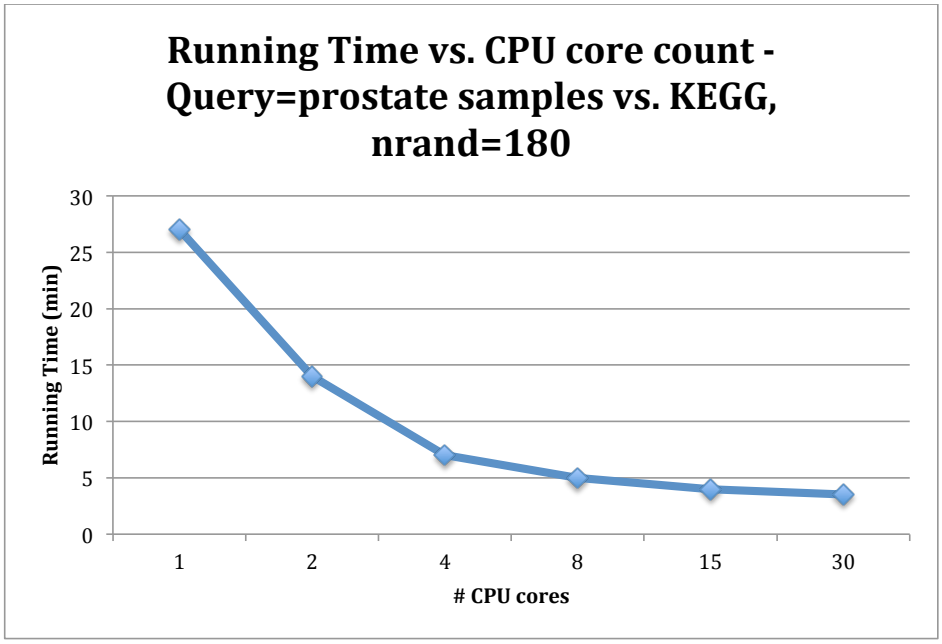
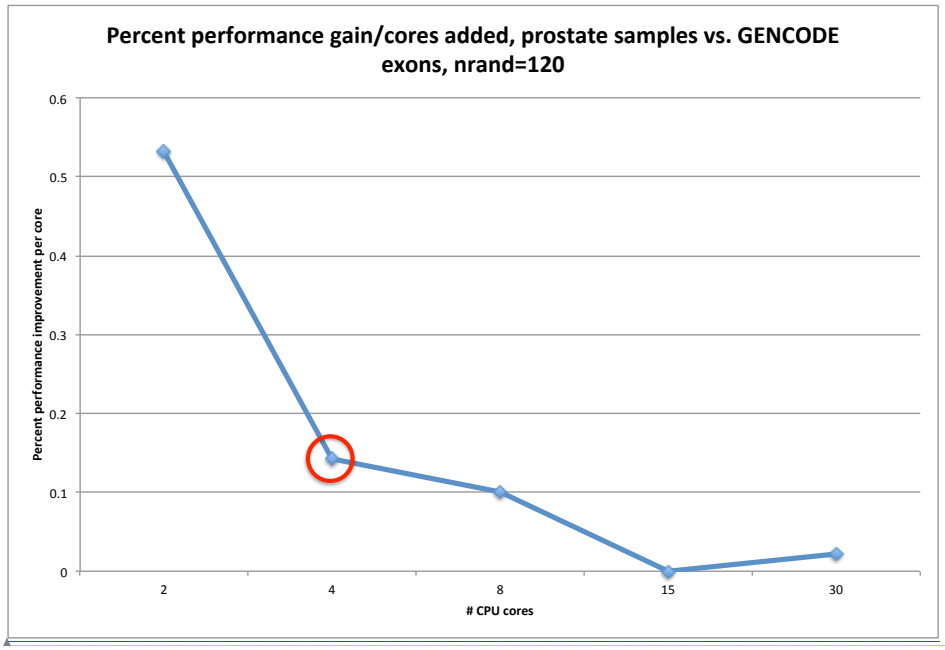


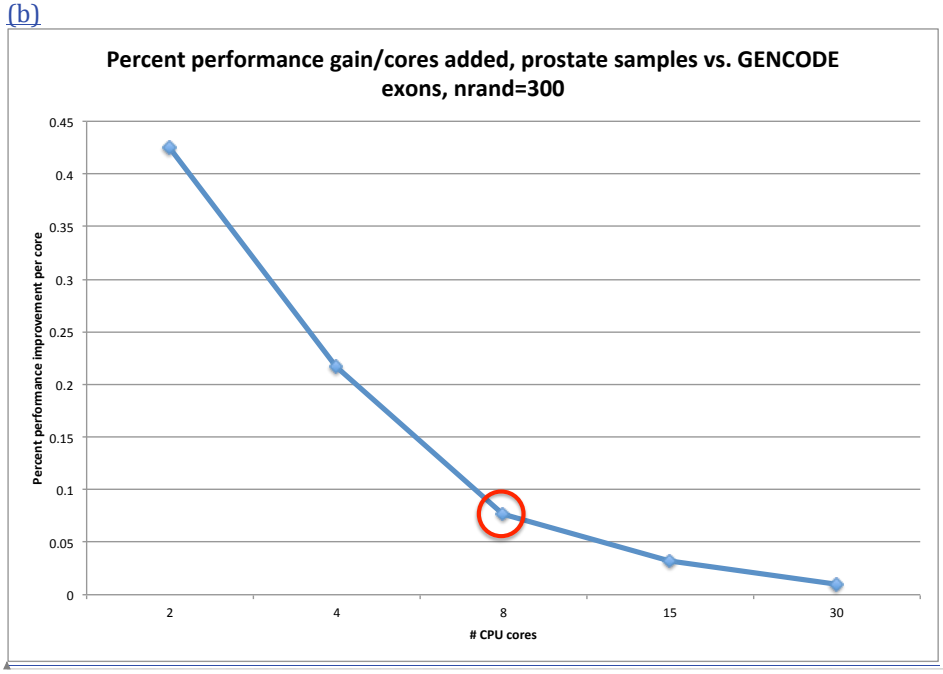
Fig. 5: These graphs illustrate the percent performance gain—measured in wallclock running time—per CPU core added. It is evident that there is a point at which the performance gained from increasing the number of parallel CPU cores diminishes sharply at a certain point, much like the variance metric in elbow plots. These are marked in red. (a) is for a LARVA-SAM analysis with the number of random datasets *nrand* set to 120, and (b) is for the same analysis with *nrand* set to 300. These graphs indicate that the "elbow" increases with *nrand*.

(a)



Unknown
Formatted: Underline

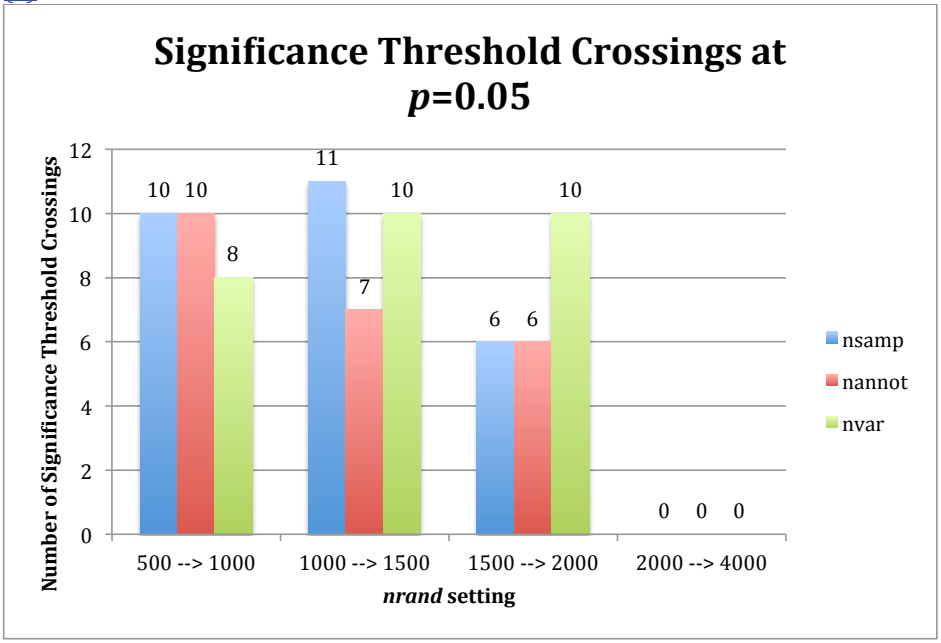
(b)



Unknown
Formatted: Underline

Fig. 6: A graph of the significance threshold crossing at p=0.05 (a) and p=0.01 (b). Significance threshold crossings refer to p-values that cross the given significance threshold by increasing the *nrand* setting as indicated in the x-axis. For 185 pathways, there are 555 p-values generated at each *nrand* setting. For both p=0.05 and p=0.01, there are no crossing after *nrand*=2000, indicating that this is the optimal *nrand* setting.

(a)



(b)

Lucas Lochovsky 4/1/14 11:41 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 11:41 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 11:59 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 4:06 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 4:05 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 12:05 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 12:05 PM
Formatted: Font:Italic

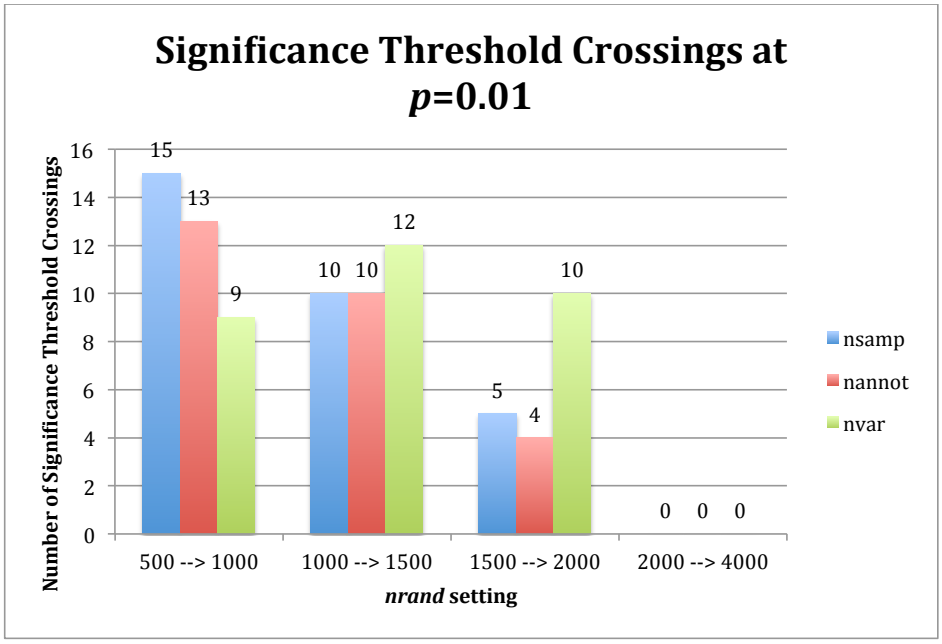
Lucas Lochovsky 4/1/14 12:05 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 12:05 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 12:05 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 12:05 PM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 11:59 AM
Formatted: No underline



Tables

Table 1: A list of the variant datasets, annotation datasets, and *nrand* settings used to test the timing of LARVA-SAM. Each combination of variant dataset, annotation dataset, and *nrand* setting was used in the timing tests in Fig. 3.

Variant Datasets		**Annotation Datasets**		***nrand* settings**
Collection of prostate cancer samples (10,356 variants)	X	GENCODE v15 exons (~191,000 annotations)	X	120
Günel grade 4 glioma set (1710 variants)		KEGG pathways' member genes (58,770 annotations)		300

Table 2: A subset of the LARVA-SAM significance test data produced by running LARVA-SAM with the prostate sample collection as the *vfiles*, and the KEGG pathway data as the *afiles*. Data was produced for *nrand* settings of 500 and 1000 (shown), as well as 1500, 2000, 4000, 6000, 8000, and 10,000 (not shown). *P*-value stability was determined by comparing the *p*-values of equivalent LARVA-SAM analyses run at different *nrand* settings. In this sample data, the *p*-value for the neuroactive ligand receptor interaction pathway changes from being insignificant at *p*=0.05 at *nrand*=500 to being significant at *nrand*=1000. We sought to find the *nrand* for

Lucas Lochovsky 3/17/14 6:59 PM
Formatted: No underline

Lucas Lochovsky 3/17/14 6:59 PM
Formatted: No underline

Lucas Lochovsky 3/17/14 7:01 PM
Formatted: Line spacing: single

Lucas Lochovsky 3/17/14 7:01 PM
Formatted: Line spacing: single

Lucas Lochovsky 3/17/14 7:01 PM
Formatted: Line spacing: single

Lucas Lochovsky 4/1/14 10:31 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:31 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:34 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:34 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:34 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:34 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:37 AM
Formatted: Font:Italic

which the *p*-values had stabilized enough that these significance threshold crossings do not occur when higher *nrand* settings are used.

nrand	Pathway	Observed nsamp	Observed nannot	Observed nvar	nsamp expected avg	nsamp *p*-value	nannot expected avg	nannot *p*-value	nvar expected avg	nvar *p*-value
500	kegg_pathways_in_cancer.txt	154	30	8	1.15E+02	2.20E-08	1.70E+01	1.78E-04	1.25E-01	1.26E-125
	kegg_focal_adhesion.txt	131	20	1	9.60E+01	6.02E-31	6.38E+00	1.30E-09	0.25	0.04163226
	kegg_neuroactive_ligand_receptor_interaction.txt	122	31	3	1.40E+02	1.88E-03	40.75	0.0573091	1.25E-01	1.76E-18
1000	kegg_pathways_in_cancer.txt	154	30	8	1.16E+02	6.60E-10	1.69E+01	6.67E-04	2.50E-01	5.26E-44
	kegg_focal_adhesion.txt	131	20	1	9.27E+01	3.09E-08	5.94E+00	2.25E-11	1.88E-01	0.01868649
	kegg_neuroactive_ligand_receptor_interaction.txt	122	31	3	1.42E+02	1.21E-03	41.8125	0.01921396	6.25E-02	3.43E-34

Lucas Lochovsky 4/1/14 10:37 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:37 AM
Formatted: Font:Italic

Lucas Lochovsky 4/1/14 10:30 AM
Formatted: No underline

Lucas Lochovsky 3/31/14 5:17 PM
Formatted: No underline