

IMPERIAL COLLEGE LONDON  
DEPARTMENT OF LIFE SCIENCES  
M.Sc. BIOINFORMATICS AND THEORETICAL SYSTEMS BIOLOGY

---

# bgwas3: a pipeline for kmer based association testing in bacteria

---

Author: Gregory Leeman  
Supervisor: Prof. Dr. Antonio Berlanga

2nd September 2019

Word count:

## Acknowledgements

I would like to thank my supervisor Antonio Berlanga for giving me the opportunity to undertake this project in which I learned to much and gained so many valuable skills for which I am sure will value in my continuing career.

# Abstract

Genome-wide association studies (GWAS) are now applied more often to microbes. However, bacteria genomes are significantly more variable in both content and sequence than eukaryotic genomes, meaning these association studies have naturally different, more complex considerations. These studies utilise short unique DNA patterns, referred to as Kmers, as units of genetic variation as apposed to single nucleotide polymorphisms. Various software solutions are now available which can perform the multiple association tests of kmers, notable the python based *pyseer*. However, there is not at present a succinct tool which pipelines the necessary pre and post data processing and analysis steps such as associating the Kmers to genes, that make an association study whole.

In metabolomic studies of bacteria there are thousands of possible measures that can be made which can be considered as traits of that bacteria at a that point in time. Coupling genomics and metabolomics may provide a powerful technique to understanding how a bacteria has adapted, and in the scope of disease, may aid in the understanding of the function of drug resistance and other virulence features. However conducting individual association tests for each trait may prove time and computationally intensive.

I developed a tool, *bgwas2*, which wraps the Kmer association function of *pyseer* with other open source software tools into a single installable package. When run, *bgwas3* takes the simplest of input files and a configuration file, and performs gene annotation, phylogeny estimation, kmer association testing, kmer-gene mapping and finally visualisation with a automatically generated web report.

In creating the tool, best practices for computational biology were exercised; resulting in a final tool that expresses appropriate testing, documentation and packaging; meaning it can easily be utilised by others in the future.

The tool was also used to test the association of 26 traits - 18 relating corresponding to a metabolomic measurement, and 8 corresponding to either a measure of antibiotic resistance or bacterial motility. In 14 of the traits, between 1 and 16450 significant Kmers were found, and were mapped to Genes.

# Contents

<b>Acknowledgements</b>	<b>1</b>
<b>Abstract</b>	<b>2</b>
<b>Abbreviations</b>	<b>5</b>
<b>List of Figures</b>	<b>6</b>
<b>List of Tables</b>	<b>6</b>
<b>1 Introduction</b>	<b>7</b>
1.1 Pangenome wide association studies . . . . .	7
1.2 The Need for Multiple Association Tests - Metabolomics . . . . .	8
1.3 Scope of work . . . . .	8
<b>2 Implementation</b>	<b>10</b>
2.1 Pipelining with CGAT-Core . . . . .	10
2.2 Genome Annotation . . . . .	11
2.3 Kmer minig and counting . . . . .	11
2.4 Phylogeny estimation and covariate generation . . . . .	11
2.5 Kmer association testing . . . . .	11
2.6 Bonferoni correction . . . . .	12
2.7 Significant Kmer mapping . . . . .	13
2.8 Visualisation . . . . .	13
2.9 Scienitif computing practices . . . . .	15
2.9.1 Testing with PyTest and Travis . . . . .	15
2.9.2 Documentation with Sphinx . . . . .	16
2.9.3 Packaging with Conda . . . . .	16

<b>3</b>	<b>Results</b>	<b>16</b>
3.1	Phenotype data preprocessing . . . . .	19
3.2	Phylogeny estimation . . . . .	19
3.3	Kmer mining . . . . .	19
3.4	Association results . . . . .	22
<b>4</b>	<b>Discussion</b>	<b>23</b>
4.1	Kmer mining . . . . .	23
4.2	Phylogeny prediction . . . . .	24
	<b>References</b>	<b>24</b>

## Abbreviations

**GWAS** Genome-wide association study

**DRMAA** Distributed resource management application

**TSV** Tab seperated value

## List of Figures

1	Pipeline of bgwas . . . . .	9
2	Graphical representation of algorithm to map Kmers to reference genes . . . . .	14
3	Correlation matrix of phenotypes . . . . .	18
4	Density plots . . . . .	20
5	Density plots . . . . .	21
6	Gens . . . . .	22

## List of Tables

1	Table of non-metabolite phenotypes . . . . .	16
2	Table of metabolite phenotypes . . . . .	17

# 1 Introduction

## 1.1 Pangenome wide association studies

Genome wide association studies have been successfully used to identify genetic and phenotype associations in a hypothesis free manner. Characteristics of microbes suggest they would be ideal candidates for GWAS: their genomes are small which limits the study size; they exude much less phenotypic flexibility; and, generally, phenotypes that are of interest such as virulence and drug resistance tend to be the result of strong selective pressures. There is also good reason to perform association studies with bacteria. As identifying the genetic basis of traits that correspond with its pathogenic ability could lead to better understanding of disease and lead to new targets for pharmaceutical intervention.

Commonly, GWAS have focused on single nucleotide polymorphisms (SNPs) as units of genetic variation. SNPs are useful in organisms where recombination is reliable and genomes are relatively constant among populations. Because of linkage disequilibrium, identifying a highly associated SNP often means that the true associated loci is nearby.

However, the nature of bacteria replication makes the use of SNPs in association studies less useful. For one, significant regions of the genome can be in linkage disequilibrium; and though some species experience high rates of recombination, the recombination is not as reliable or consistent as in eukaryotes to reliably reduce linkage.

Secondly, bacteria experience much higher trans- during recombination. Therefore, bacteria of the same species can vary considerably in both order of gene content, and in the accessory genome - regions of the genome which differ.

Therefore, first finding SNPs becomes a tricky process which involves multiple alignments. Such tools such as MUMmer (and other?) do meet this problem, but even still, SNPs associated with only the core genome can be retrieved and does not consider the huge variable accessory genome in bacteria/ will not encapsulate the full

As an alternative, some recent studies have instead utilised K-mers.

Kmers are simply variable length sequences of DNA. Initially a concept of genome assembly. Can capture multiple genetic variations SNPs, longer deletions/ insertions and recombination. The size of kmers affects the genetic variation captured. Longer more specific. Shorter are more sensitive.

Getting kmers is alignment free, releases the burden of full alignment. But more specifically utilising kmers



means the core and accessory genome can be tested.

*Seer* (Lees et al. 2016) and its python reimplementation *pyseer* (Lees et al. 2018) *dbgwas*

## 1.2 The Need for Multiple Association Tests - Metabolomics

Metabolomics provides a detailed snapshot of an organism's physiological state through the quantification of hundreds to thousands of small molecules (Zampieri et al., 2017).

Coupling genomics and metabolomics to study *Pseudomonas aeruginosa* in unique clinical samples can allow us to understand how bacteria adapt to the lung environment.

The nature of metabolomics means that multiple molecules abundance may be measured and of interest to investigate.

Determine the genetic basis of metabolomics of *P.* in the context of chronic infections

## 1.3 Scope of work

The foundation of this project was to create a tool, *bgwas3*, an integration of multiple open source genomics tools and custom scripts written in python, R and bash into a single installable package that facilitates conducting multiple pangenome wide association studies in a single user step.

The tool aims to be comprehensive and robust enough to convert only the most basic required input files into valuable and interpretable results including static and interactive visualisations.

When run on a node within a computer cluster, tasks that are computationally intensive or long are distributed among the cluster, and can run simultaneously.

Through the additional configuration file, the specifics of multiple intermediary steps can be altered, which can have variable effects on the final results.

In light of the necessity to perform multiple.. (1)

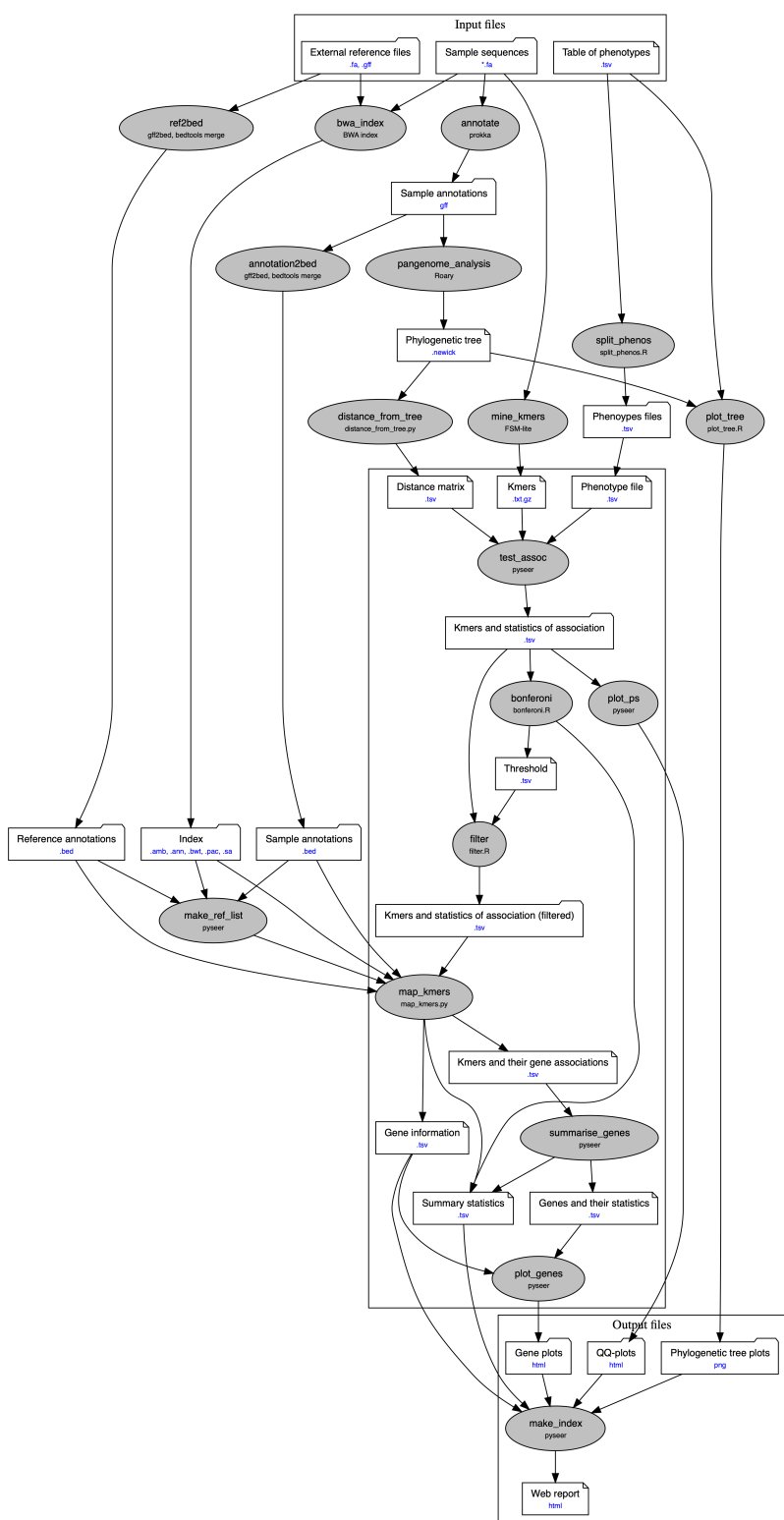


Figure 1: Pipeline of bgwas

## 2 Implementation

### 2.1 Pipelining with CGAT-Core

I made *bgwas3* with the pipelining framework CGATCore, (Cribbs et al. 2019). CGATCore is extremely portable framework that comes as a dependency free Python package, but has been used to construct complex pipelines that are scalable to large amounts of data.

With the framework, individual pipelining steps, referred to as ‘tasks’, are defined as python functions whose arguments include the necessary input files and the expected output files. With the use of python decorators, the output files of one task can be used as the input arguments of later tasks. This allows multiple data analysis to be strung together and run in the correct order. The framework utilises a system of checking the modification date of files, so only runs tasks in which their are either new input files, or input files modified. This checking system means a directory of project files may move from one location to another, even between computers and the framework will still recognise which tasks and steps are out of date.

Another significant feature of CGATCore which makes it stand out amongst other pipelining tools is an interface to control a distributed resource management application API (DRMAA) such as PBS-Pro used by Imperial College London’s resource computing service. Tasks which process files in parallel are large can run in parallel as batch processes, and tasks which require either high storage or memory resources it can be distributed to nodes which match these requirements. This makes executing pipeline such as *bgwas3*, which involve multiple computationally intensive intermediary steps faster.

*bgwas3* was written as 18 ‘tasks’ which are visualised in figure 1.

I wrote the pipeline in python, but individual tasks made calls to external tools or scripts I wrote in either python, R and bashscript.

As input, the tool requires the following three starting files:

1. A directory named ‘contigs’ containing fasta files uniquely named for each bacteria sample
2. A tab separated value file (TSV) in which the first column matches the names of sample fasta files, and remaining columns correspond to phenotype values.
3. An optional directory of reference sequences in fasta format accompanied by annotation files in gff format.

## 2.2 Genome Annotation

In the *bgwas* task ‘annotatate’ all sample sequences are annotated by the external tool Prokka (Seemann 2014) Prokka first makes predictions on features using a selection of external tools including Prodigal (Hyatt et al. 2010) for the anotation of coding region, RNAmmer (Lagesen et al. 2007) which identifies for ribosomal RNA genes, Aragorn (Laslett and Canback 2004) for transfer RNA genes, and SignalP (Armenteros et al. 2019) for signal leader peptides and Infernal for non-coding RNA. After feature prediction, the speculative features are queried against a umber of databases including UniProt [@], Refseq [@] and Pfam.

Draft sequences are annotated in the *bgwas3* pipeline for two reasons. First, gene annotations are later used to estimate the phylogenetic tree of the sample, and secondly, significantly associated Kmers are later mapped to the annotated genomes when attempting identifying the Kmer’s genetic identity.

## 2.3 Kmer minig and counting

*bgwas3* integrates the external tool FSM-lite (Välimäki 2018) to ‘mine’ and count Kmers. A benefit of FSM-lite is, unlike other Kmer mining tools such as DSK (Rizk, Lavenier, and Chikhi 2013), FSM-lite allows a range of kmer-sizes, as defined by the user, to be mined as apposed to a single length. *bgwas3* allows the Kmers length to be changed in the pipeline by editing the configuration yml file values ‘fsm\_kmer\_min’ and ‘fsm\_kmer\_max’.

## 2.4 Phylogeny estimation and covariate generation

*bgwas3* currently takes a pangenomic approach to phylogeny estimation; as in the relative prescence and absence of genes are the considering factor which the distance between samples are determined. An phylogenetic tree is estimated with the tool Roary (Page et al. 2015). In summary, Roary determnes genes which fall within the coregeneome, and then peforms clustering of isolates based in the constetution of the variable accessory geneome. One of roary’s outputs, a tree in the common newick format, which is then converted into a distance matrix tsv file using a python script.

The single reasoning for prediciting phylogeny in the *bgwas3* pipeine is to use the distances defined in the distance matrix as covariates in the later association testing.

## 2.5 Kmer association testing

The typical analytical strategy GWAS involves some form of linear regression.

It is almost always wrong to assume that a trait is the only.

If a phenotype is linked with ancestry, then other genetic polymorphism may be incorrectly identified as linked to the trait of interest.

Many methods are deployed in an attempt to reduce the inevitable effect of population structure has on association studies.

In human studies, for example, it's not abnormal to remove highly related individuals, though this will ultimately reduce the power of the study by decreasing sample size.

Another popular technique involves performing some multiple dimensional scaling, and then including significant principal components as fixed effects in the model.

*bgwas* instead attempts to tackle the problem of population structure by one of the three available tests of association available in the *pyseer* (Lees et al. 2018) suite: linear mixed model (LMM), specifically factored spectrally transformed as LMM (Lippert et al. 2011).

LMMs can capture all of these confounders simultaneously, without knowledge of which are present and without the need to tease them apart.

Roughly speaking, LMMs tackle confounders by using measures of genetic similarity to capture the probabilities that pairs of individuals have causative alleles in common. Such measures include those based on identity by descent.

Given a means of determining the hierarchical relatedness between samples, the mixed model is generally preferred, and has been shown in past studies to control the inflation of p-values better (Lees et al. 2017).

*bgwas* uses the distance matrix from the estimated phylogeny as covariates.

In the *bgwas* task 'test\_assoc' and association test for each phenotype on all the Kmers. In a computer cluster, these individual tests may be run simultaneously.

## 2.6 Bonferroni correction

The output of *pyseer*, a list of all kmers and statistics relating to their association to the given phenotype, are then filtered by their p-value through bonferroni correction.

This is done in two steps. First the bonferroni threshold is calculated from the number of kmers minded.

*Bgwas3* allows the user to change the alpha level

The Bonferroni correction is a multiple-comparison correction used when several dependent or independent statistical tests, utilised to limit the number of spurious positive tests. If multiple hypotheses are tested, the chance of a rare event increases, and therefore, the likelihood of incorrectly rejecting a null hypothesis (i.e., making a Type I error) increases.

Only significant kmers are then used in later steps of the pipeline.

## 2.7 Significant Kmer mapping

The Burrows-Wheeler Alignment Tool (BWA) is primarily used to map significant Kmers to genes.

The tool first requires that sequences are first converted into an FM-index, a data-structure similar to a suffix array. This is performed with the *bgwas* task 'bwa\_index', on all sample and reference fast files.

For use with bedtools, all annotation files in gff are similarly converted into bed format, and then subsequently filtered with the tasks 'annatation2bed' and 'ref2bed' respectively. Sample annotation from prokka are filtered to include only those annotations which correspond to a named gene, and so exclude ones which relate to hypothetical proteins or non-gene loci. Reference annotations which correspond to the same region but are present as two separate entries are merged, keeping the information from both, as this was found to be a common nomenclature in well annotated references.

The task 'map\_kmers', executes a python script I wrote whose algorithm is visualised in ??.

In summary, the algorithm works as follows: - A fake multi-fasta file is generated with an entry for each unmapped, significant kmer - The next reference is chosen - The command-line tool 'bwa-mem' (Li 2013) attempts to align to the reference - A fake bed file is generated with an entry corresponding to each successful alignment with bwa mem - The query bed file is compared to the reference bed file using the command line tool 'bed tools intersect' - The information about each intersection is harvested and stored in a gene info file, while the kmer is marked as mapped Repeat until all kmers mapped or until all references have been used

## 2.8 Visualisation

An important feature of *bgwas3* is the automated generation of multiple figures, integrated into a final web-based report.

Static visualisations are made with external scripts I wrote in R that make use of ggplot2 package (??).

The task 'plot\_ps' generated a quantile-quantile plot from all p-values unfiltered. This may give See fig

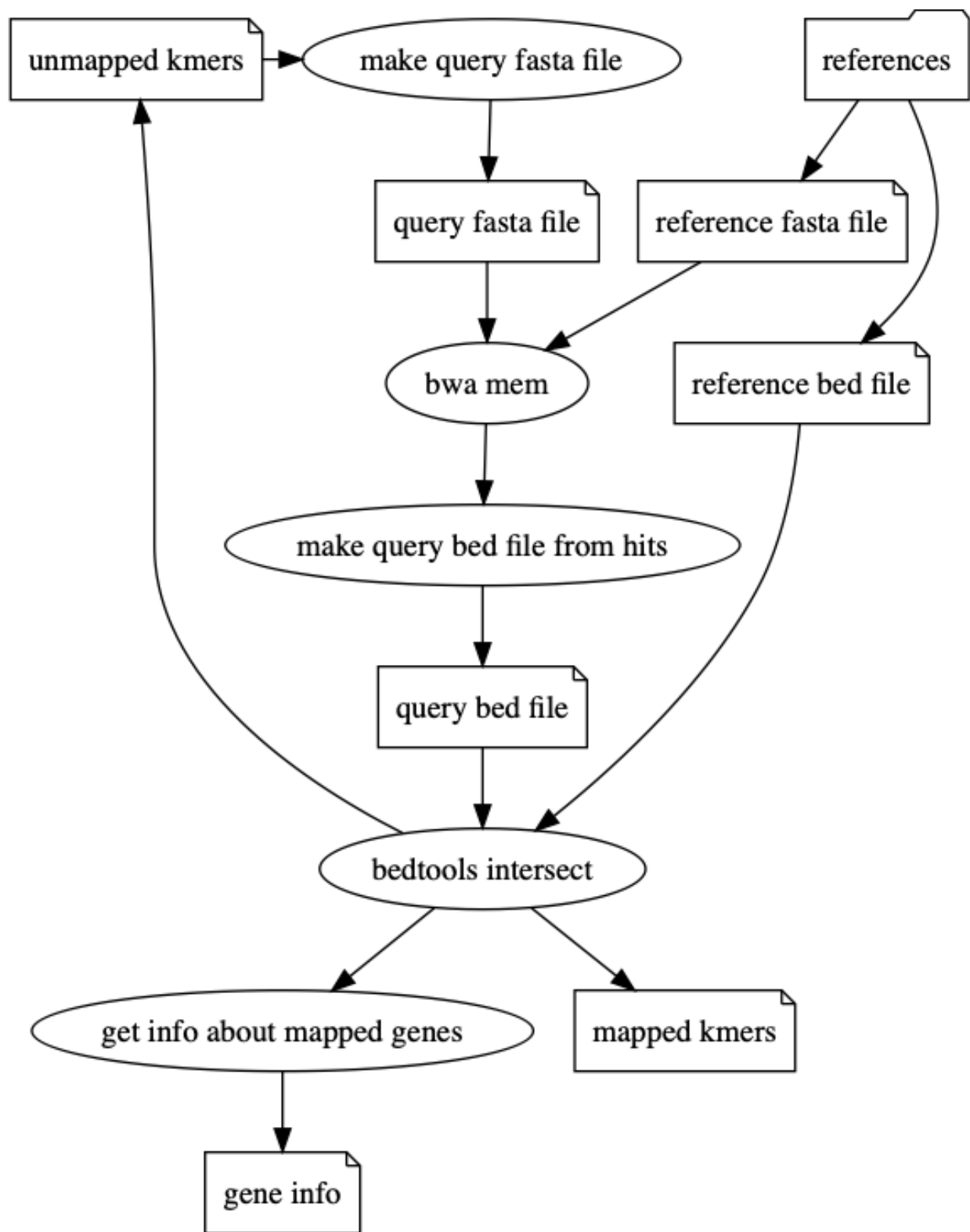


Figure 2: Graphical representation of algorithm to map Kmers to reference genes

A single phylogenetic tree is built from the newick file and phenotype file utilising external R package (???).  
See figs

Finally, for each phenotype, a plot of genes is made in which the following features are visually encoded

- Maximum  $-\log_{10}(\text{p-value})$  of Kmers mapped to that gene
- Average beta value (effect size)
- Average allele frequency
- Number of kmers ‘hits’

Finally, a web based report which incorporates all the static visualisations, and remakes an interactive visualisation analogous to the gene plot, is built with the task “make\_index”.

A datatable which can be filtered by number of genes

The interactive visualisation for each gene Shows

## 2.9 Scientific computing practices

A significant goal of bgwas3 was to implement a useful and reusable tool that may be used outside the scope of this singular project.

As such, various software development principles and concepts were applied to the project

CGAT-core is implemented in Python 3 and installable via Conda and PyPI with minimal dependencies. We have successfully deployed and tested the code on OSX, Red Hat and Ubuntu. We have made CGAT-core and associated repositories open-source under the MIT licence, allowing full and free use for both commercial and non-commercial purposes. Our software is fully documented (<https://pypi.org>), version controlled and has extensive testing using continuous integration (<https://travis-ci.org/cgat-developers>.) We welcome community participation in code development and issue reporting through GitHub

### 2.9.1 Testing with PyTest and Travis

Pytest and



### 2.9.2 Documentation with Sphinx

### 2.9.3 Packaging with Conda

## 3 Results

**Dataset** For the development, testing, and evaluation of *bgwas3* a dataset of genomes and phenotypes related to *Pseudomonas aeruginosa* was used.

Cystic fibrosis is an autosomal recessive disorder that, due to a single gene mutation, leads to a defective trans-membrane regulator protein. This protein is situated in epithelial cells that make up the mucus membranes of the body, and is primarily responsible for transporting chloride ions and bicarbonate to these membranes. The dysfunction of this protein ultimately limits the osmotic movement of water, and these membranes remain viscous and immotile. In the lungs of healthy individuals a less viscous mucus is able to be transported by cilia out of the lungs. In patients with CF, the mucus instead becomes an ideal environment for bacteria to propagate, and so CF patients experience at first episodic, but then chronic infection of the lungs. Even with prolonged antibiotic use, the fast generation time of bacteria means resistant strains soon develop, and the prolonged inflammation leads to respiratory failure and death.

*Pseudomonas aeruginosa* is a gram negative bacteria which can easily integrate exogenous DNA into its own genome, making it able to adapt to antibiotic pressures rapidly. For this reason, it is a common hospital-acquired infection. It is also one of the primary bacteria present in the lungs of late stage and terminal patients with cystic fibrosis when the lungs function starts to decrease. For this reason, understanding the genetic adaptations *Pseudomonas* experience when in chronic state is significantly important in tackling this disease.

In a previous study (Behrends et al. 2013) 91 strains were collected over a period of 24 years from 18 patients suffering from Cystic Fibrosis.

*bgwas3* was used to test the genetic association of 26 traits (see tables 1 and 2) 18 relating to a metabolomic measurement and 8 corresponding to either a measure of antibiotic resistance or bacterial motility.

Table 1: Table of non-metabolite phenotypes

Name	Description
Tobromycin	Resistance to inhaled antibiotic Tobromycin

Name	Description
Imipenem	Resistance to intravenous antibiotic Imipenem
Aztreonam	Resistance to intravenous/intramuscular antibiotic Aztreonam
Ciprofloxacin	Resistance to oral antibiotic Ciprofloxacin
Colistin	Resistance to ‘last-resort’ antibiotic Colistin
Swim	Measure of cell surface bacteria movement by flagella
Swarm	Measure of rapid surface movement by multiple bacteria with rotating flagella
Twitch	Measure of slow bacteria movement powered by pili

Table 2: Table of metabolite phenotypes

Chemical
Hydrogen Cyanide
Cyanide
2-Furoate
3-Hydroxyisovalerate
3-Methylthiopropionic acid
Anthranilate
Betaine
Cystine
Formate
Fumarate
Histidine
Isoleucine
Leucine
Methanol
Methionine
Tryptophan
Uracil
Valine

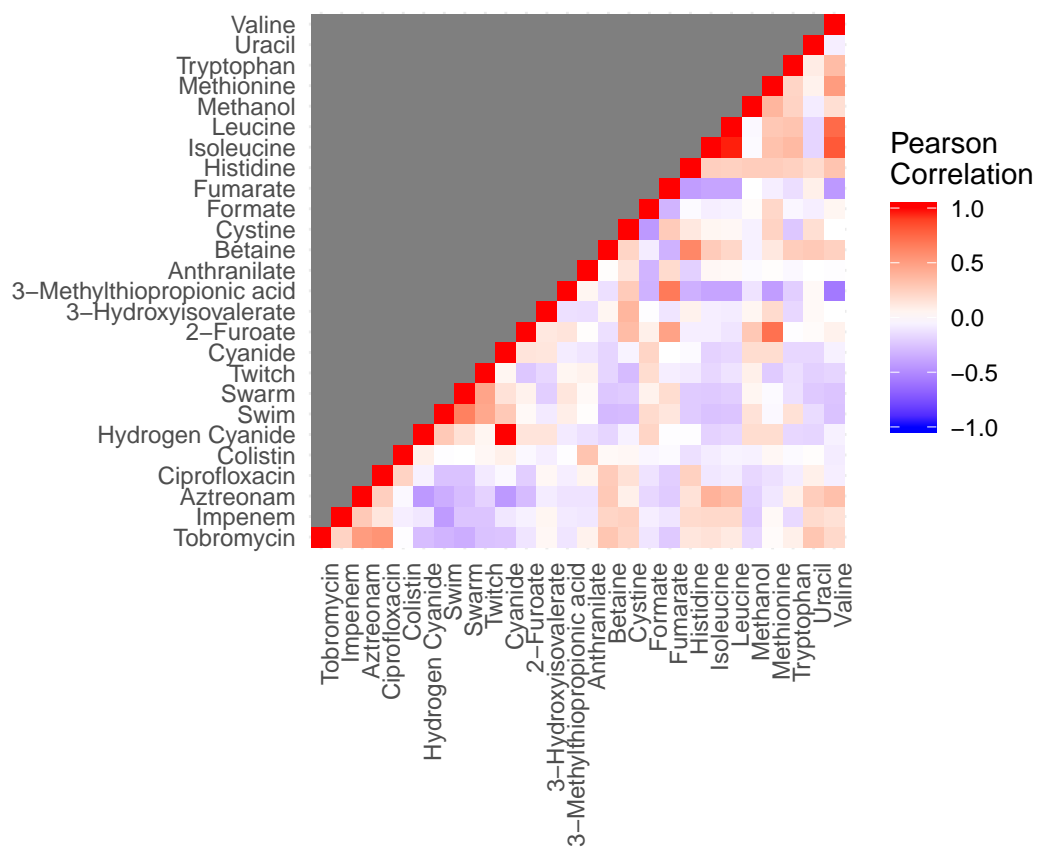


Figure 3: Correlation matrix of phenotypes

From the correlation matrix, it appears that most phenotypes corresponding to antibiotic resistance seem to be somewhat positive correlated, but similarly negatively correlated to measures of motility. Strong levels of positive correlation were found between metabolites valine, with isoleucine and isoleucine.

### 3.1 Phenotype data preprocessing

When association testing a qualitative variable with linear regression, it is generally assumed that the variable follows a normal distribution. When this assumption proves not to be true, and the continuous trait displays severe skewness the regression may fail to control for type-1 errors (ref)

When a continuous trait is non-normal, a popular statistical technique in GWAS involves transforming the data. Often, a simple log transformation can be sufficient. Recently, rank based inverse normal transformations (INT) have become popular among genetics researchers.

Prior to running *bgwas*, the 26 phenotypes were separately log transformed and INT, essentially tripling the number of phenotypes tested to 78. Density plots of the unadjusted and transformed traits are visualised in figure 4.

### 3.2 Phylogeny estimation

All 91 genomes were annotated and 14643 unique genes were identified with Prokka. As identified by Roary, these genes were found in >99% of the genomes, and constitute the core genome, leaving 10005 in the accessory.

From these genes, a phylogenetic tree was estimated, and visualised (figure 5).

The patient of which samples were sourced from are visually encoded by the color of the tooltips. An inspection of the tree shows that, in general, samples from the same patient are generally clustered together, which leads me to believe the phylogenetic estimate is not completely untrue. It is not unexpected that the dividing between patients is not perfect, due to possible transfection.

Traits of antibiotic resistance seem to be best associated with phylogeny, as seen in fig.

### 3.3 Kmer mining

497827 unique kmers were mined of sizes between 9 and 100 base pairs in length.

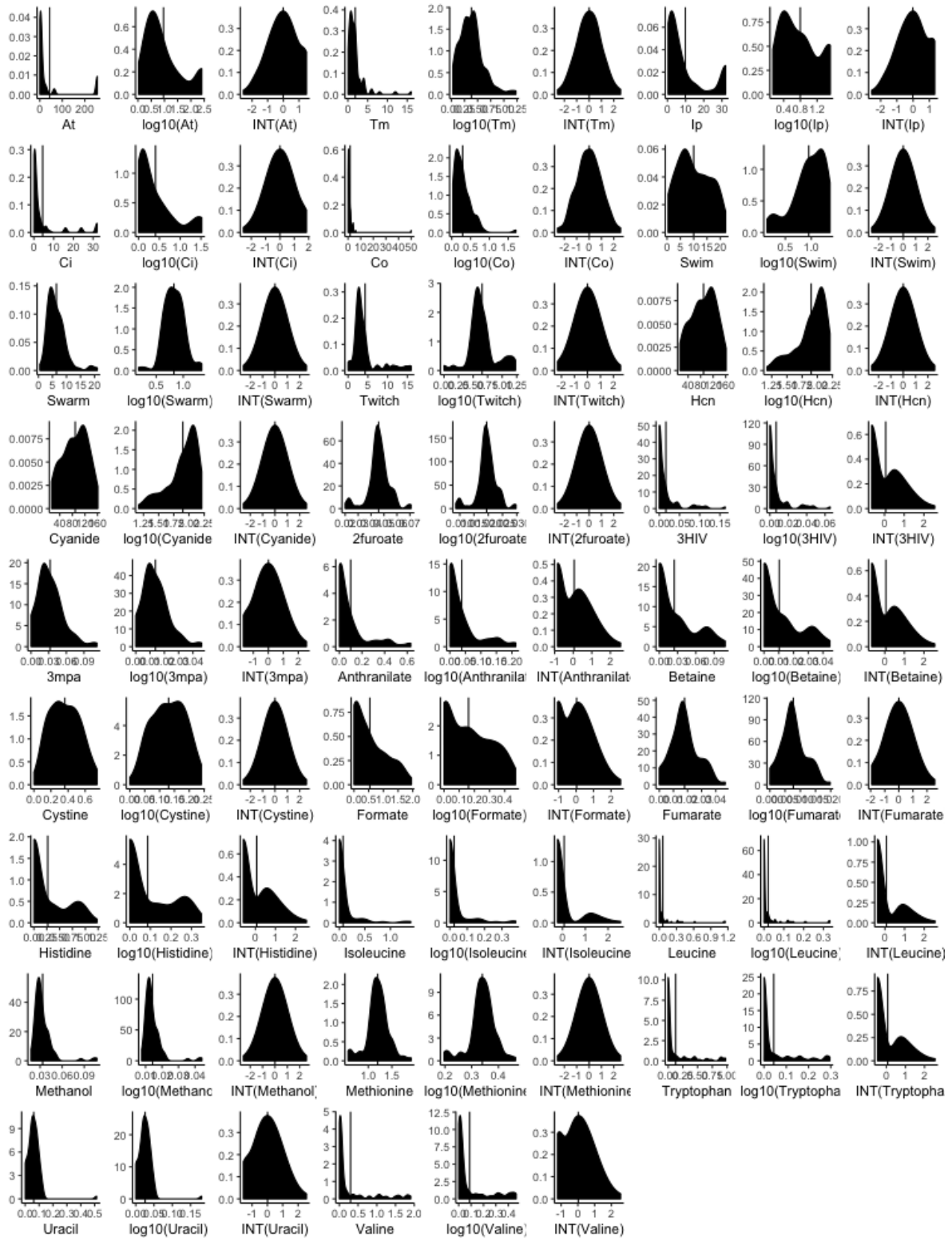


Figure 4: Density plots

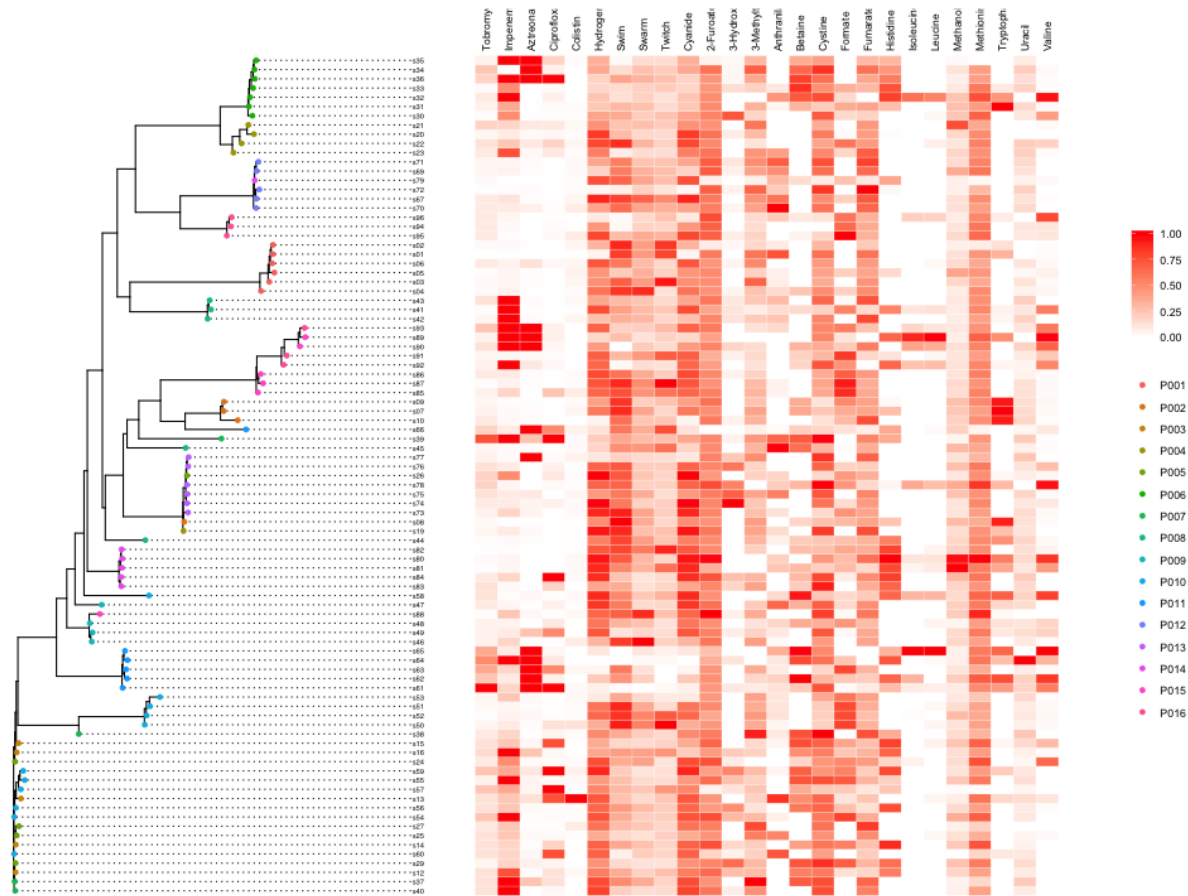


Figure 5: Density plots

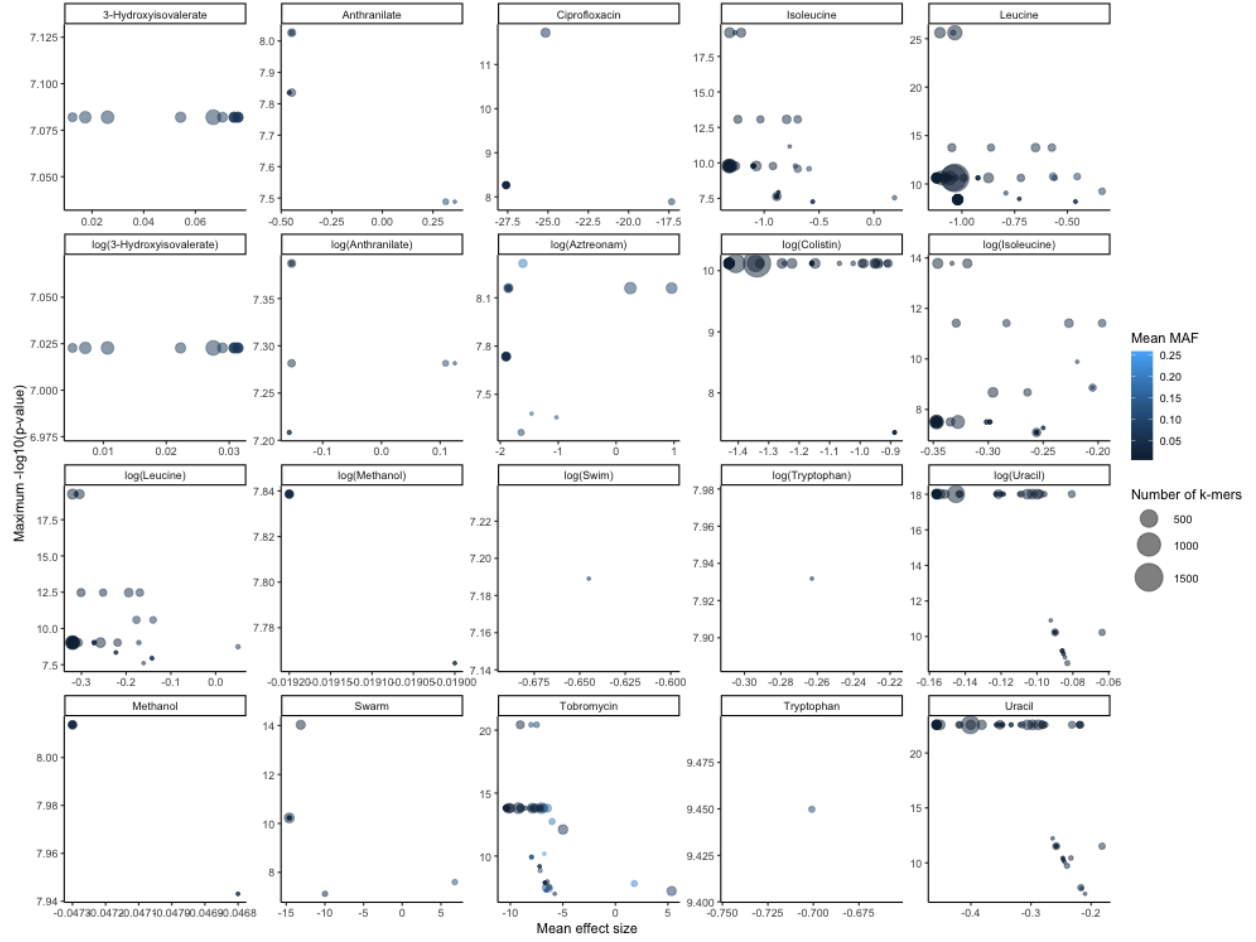


Figure 6: Gens

### 3.4 Association results

14 of the 26 traits had significant Kmers that passed the bonferoni threshold of  $p < 1.004 \cdot 10^{-7}$ . Of the 26 traits 14 of the traits, between 1 and 16450 significant Kmers were found, and were mapped to Genes (see ??). The trait with the most number of signifiant kmers dound was the one corresponding to Leucine, in which

Phenotype	Significant Kmers	Genes
Leucine	16450	289
log(Colistin)	9811	233
Isoleucine	5097	132
log(Isoleucine)	4985	128
log(Leucine)	4913	125

Phenotype	Significant Kmers	Genes
Uracil	2707	75
log(Uracil)	2470	72
Tobromycin	1848	46
log(Aztreonam)	900	19
Ciprofloxacin	285	14
3-Hydroxyisovalerate	1814	13
log(3-Hydroxyisovalerate)	1814	13
Anthranilate	210	7
log(Anthranilate)	210	7
Methanol	216	7
log(Methanol)	216	7
Swarm	229	7
log(Swim)	2	1
Tryptophan	20	1
log(Tryptophan)	1	1
Aztreonam	26891	0
log(Ciprofloxacin)	1	0
log(Hydrogen Cyanide)	1	0
isoleucine__int	1	0

## 4 Discussion

In this work... ## Genome annotation

### 4.1 Kmer mining

The tool fsm

Prokka is a good tool ?Settings



## 4.2 Phylogeny prediction

Currently, *bgwas3* implements only a pangenomic approach approach of distance estimation. There are other tools which involve alignment of the core genome and snps. . . May or may not be better Reintroduce the problem of a large multiple alignment.

The strong LD caused by the clonal reproduction of bacterial populations means that non-causal k-mers may also appear to be associated.?

## References

- Armenteros, José Juan Almagro, Konstantinos D. Tsirigos, Casper Kaae Sønderby, Thomas Nordahl Petersen, Ole Winther, Søren Brunak, Gunnar von Heijne, and Henrik Nielsen. 2019. “SignalP 5.0 Improves Signal Peptide Predictions Using Deep Neural Networks.” *Nature Biotechnology* 37 (4): 420–23. <https://doi.org/10.1038/s41587-019-0036-z>.
- Behrends, V., B. Ryall, J. E. A. Zlosnik, D. P. Speert, J. G. Bundy, and H. D. Williams. 2013. “Metabolic Adaptations of *Pseudomonas Aeruginosa* During Cystic Fibrosis Chronic Lung Infections: Metabolomic Adaptations of Cystic Fibrosis Isolates.” *Environmental Microbiology* 15 (2): 398–408. <https://doi.org/10.1111/j.1462-2920.2012.02840.x>.
- Cribbs, Adam P., Sebastian Luna-Valero, Charlotte George, Ian M. Sudbery, Antonio J. Berlanga-Taylor, Stephen N. Sansom, Tom Smith, et al. 2019. “CGAT-Core: A Python Framework for Building Scalable, Reproducible Computational Biology Workflows.” *F1000Research* 8 (April): 377. <https://doi.org/10.12688/f1000research.18674.1>.
- Hyatt, Doug, Gwo-Liang Chen, Philip F. Locascio, Miriam L. Land, Frank W. Larimer, and Loren J. Hauser. 2010. “Prodigal: Prokaryotic Gene Recognition and Translation Initiation Site Identification.” *BMC Bioinformatics* 11 (March): 119. <https://doi.org/10.1186/1471-2105-11-119>.
- Lagesen, Karin, Peter Hallin, Einar Andreas Rødland, Hans-Henrik Staerfeldt, Torbjørn Rognes, and David W. Ussery. 2007. “RNAmmer: Consistent and Rapid Annotation of Ribosomal RNA Genes.” *Nucleic Acids Research* 35 (9): 3100–3108. <https://doi.org/10.1093/nar/gkm160>.
- Laslett, Dean, and Bjorn Canback. 2004. “ARAGORN, a Program to Detect tRNA Genes and tmRNA Genes in Nucleotide Sequences.” *Nucleic Acids Research* 32 (1): 11–16. <https://doi.org/10.1093/nar/gkh152>.

- Lees, John A, Nicholas J Croucher, David Goldblatt, François Nosten, Julian Parkhill, Claudia Turner, Paul Turner, and Stephen D Bentley. 2017. “Genome-Wide Identification of Lineage and Locus Specific Variation Associated with Pneumococcal Carriage Duration.” Edited by Sarah Cobey. *eLife* 6 (July): e26255. <https://doi.org/10.7554/eLife.26255>.
- Lees, John A, Marco Galardini, Stephen D Bentley, Jeffrey N Weiser, and Jukka Corander. 2018. “Py-seer: A Comprehensive Tool for Microbial Pangenome-Wide Association Studies.” Edited by Oliver Stegle. *Bioinformatics* 34 (24): 4310–2. <https://doi.org/10.1093/bioinformatics/bty539>.
- Lees, John A., Minna Vehkala, Niko Välimäki, Simon R. Harris, Claire Chewapreecha, Nicholas J. Croucher, Pekka Marttinen, et al. 2016. “Sequence Element Enrichment Analysis to Determine the Genetic Basis of Bacterial Phenotypes.” *Nature Communications* 7 (1): 12797. <https://doi.org/10.1038/ncomms12797>.
- Li, Heng. 2013. “Aligning Sequence Reads, Clone Sequences and Assembly Contigs with BWA-MEM.” *arXiv:1303.3997 [Q-Bio]*, March. <http://arxiv.org/abs/1303.3997>.
- Lippert, Christoph, Jennifer Listgarten, Ying Liu, Carl M Kadie, Robert I Davidson, and David Heckerman. 2011. “FaST Linear Mixed Models for Genome-Wide Association Studies.” *Nature Methods* 8 (10): 833–35. <https://doi.org/10.1038/nmeth.1681>.
- Page, Andrew J., Carla A. Cummins, Martin Hunt, Vanessa K. Wong, Sandra Reuter, Matthew T. G. Holden, Maria Fookes, Daniel Falush, Jacqueline A. Keane, and Julian Parkhill. 2015. “Roary: Rapid Large-Scale Prokaryote Pan Genome Analysis.” *Bioinformatics* 31 (22): 3691–3. <https://doi.org/10.1093/bioinformatics/btv421>.
- Rizk, Guillaume, Dominique Lavenier, and Rayan Chikhi. 2013. “DSK: K-Mer Counting with Very Low Memory Usage.” *Bioinformatics (Oxford, England)* 29 (5): 652–53. <https://doi.org/10.1093/bioinformatics/btt020>.
- Seemann, Torsten. 2014. “Prokka: Rapid Prokaryotic Genome Annotation.” *Bioinformatics* 30 (14): 2068–9. <https://doi.org/10.1093/bioinformatics/btu153>.
- Välimäki, Niko. 2018. “Fsm-Lite.” <https://github.com/nvalimak/fsm-lite>.