Metagenomics of Acid Soil
a study of Nanopore long-reads and Acidobacteria

The production of **acidoseq**, a Python package designed to observe the Acidobacteria sequences from a soil sample. Studying the GC ratio of the unclassified reads in order to predict their subdivision based on the pH of the soil.

Author

Samantha Pendleton BSc

Student ID: **140159202**

Supervisor

Dr Amanda J. Clare

Dissertation in fulfilment of Masters of Science degree

Data Science MSc
CSM9060

Computer Science
Aberystwyth University

September 26, 2018

Declaration

I declare that all work provided in this Dissertation is my, Samantha Pendleton, own work and taken upon my own personal responsibility, unless otherwise stated. Work was under the supervision of Amanda Clare.

I understand the consequences and the penalties in results of plagiarism, that can lead to loss of marks - I have read through the student handbook and understood unfair practices. I agree that my work can be available for students, academic staff, and researches.

The research and documented information is in fulfilment of Data Science MSc, Masters of Science, degree - topic focused on bioinformatics/computational biology.
The data obtained were retrieved from the *Institute of Biological, Environmental and Rural Sciences* at Aberystwyth University.

Ethical clearance reference: **10584**; approved at a departmental level of Computer Science - I understand the university's guidelines and have complied with them. Please see the AU Research Ethics Framework for further details - with queries, contact: *ethics@aber.ac.uk*.

X 

Samantha Pendleton

26/09/2018

Date

Acknowledgements

I would like to spend a little time expressing my appreciation for the many individuals who brought me here today - I wouldn't have completed my Masters dissertation, and degree, without them.

Firstly, my supervisor Amanda Clare, whom I have worked with in the past. My experience with Amanda influenced me to work with her again since she helped me find my interest in bioinformatics through her supportive leadership. Amanda continuously motivated me throughout my work; I want to voice that I appreciate and very grateful of her enthusiasm and innovative teaching/guidance.

Edel Sherratt, my tutor, mentored me during my undergraduate degree in addition to my Masters: I am extremely thankful to have Edel consistently provide aid & guidance, she had an eager approach to work and would always offer her support.

The data was generated by Arwyn Edwards, Aliyah Debbonaire, and André Soares - I want to express gratitude for providing the opportunity for intellectual engagement. I have worked with both Edwards and Soares in the past and once again proved enjoyment for metagenomes.

I would like to show recognition for my other module lecturers - Kim Kenobi for reintroducing me to statistics: I found my interests in epidemiology plus transcriptomics; Chuan Lu for developing my knowledge further in machine learning plus databases; and David Hunter & Yonghuai Liu for allowing me to understand the difficulties in mining complicated data-sets.

Additionally, my family, friends, and fellow Masters colleagues had cheered me on throughout my studies. I would like to thank soil for existing, without soil I could not have created this dissertation and discovered some interesting and dirty facts[1].

Finally, a thank you to, Sam Nicholls. You endlessly dispensed encouragement. Thank you, Dr Nicholls, for reliably supporting me through my undergraduate, research projects, and my Masters' dissertation. I hope he resumes the support during my PhD and *our* next adventure.

This Masters degree has been such an excellent experience: I have learnt so many skills this year and it has helped me pursue life goals. I wish to show acknowledgement for all those who helped in any way, their confidence in me helped the chase for my ambitions.

[1] *pun intended*

Abstract

We obtained a data-set from a soil sample in Aberystywth with the aim to use computational methods, such as collating groups of bacteria and merging parts of the data together, to observe the diversity of species. We aimed to scope techniques to study the biology of the data and obtain a look into this new type of data format.

Furthermore, Acidobacteria is a newly discovered group of bacteria and found in soils. Acidobacteria is majority full of unclassified species. We read about some patterns that emerge within Acidobacteria's inner groups and we found that these inner groups also depend on pH of soils. Our project was aimed at linking these patterns together and proving there is an underlying pattern. Also, we want to observe this pH dependency of Acidobacteria groups to prove group similarity.

The project principle was creating a package with a programming language most suited with various third-party libraries in order to observe a particular data-set, more specifically the portion of which are assigned Acidobacteria to extract information.

Our package worked as expected, we used another tool first and their output file became useful. We found that the patterns in the Acidobacteria groups were consistent and exploring these results we found species in the groups that should appear. Looking at the methods to observe the diversity of the soil, we found Acidobacteria present with all the various tools, however they did not work as expected with this new data.

We concluded that perhaps this new generation of data still needs to be understood and looked into more deeply, plus we found that lab techniques in order to retrieve this data is still quite sensitive and our data, despite promising results from the package, could have benefited if a bigger and more reliable set was available.

Word count 11,839

Version v4.41

Contents

List of Figures

List of Tables

List of Listings

Part I

Introduction

Chapter 1

Introduction

1.1 Computational Biology & Bioinformatics

Computational biology and bioinformatics are very similar disciplines. Computational biology is using computational methods to study biology, whilst bioinformatics is creating solutions in order to solve biological problems. Both methods are analysing biological data-sets, including sequences [1].

DNA *Deoxyribo Nucleic Acid* is a coiled module containing the genetic material of the human genome: the complete set of genes or genetic material present in a cell or organism [2]. This coiled module contains two strands, polynucleotides, which are composed of units called nucleotides [3]. Each nucleotide is composed of one of four nitrogen-containing nucleobases: adenine `A`, cytosine `C`, guanine `G`, thymine T. These are bases and each base creates a base-pair *bp*: `GC` and `AT` are base-pairs [4]. See the structure of DNA in figure 1.1.

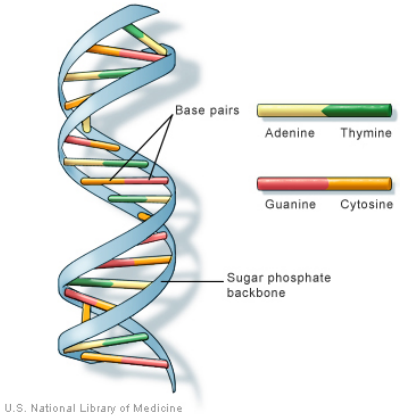


Figure 1.1: Structure of DNA. Image credit: *U.S. National Library of Medicine*.

Metagenomics is the study of genetic material retrieved from environment samples - understanding the diverse species which reside in these microbial environments [5]. This research area applies tools, produced from the computational biology research areas, to directly access the genetic content of these bio-diverse communities [6].

In this dissertation, I am analysing soil metagenomes, specifically, Acidobacteria. I am using computational languages and biological methods in order to study the biology and data-set (diversity of Aberystwyth soil), plus produce a tool useful for the analysis of Acidobacteria `AGCT` content and other features.

1.2 Aberystwyth

Aberystwyth is a town on the west coast of Wales (situated mid-west), located near the confluence of the rivers *Ystwyth* and *Rheidol*. The Rheidol flows through the town, whilst Ystwyth flows on the outskirts.

In Ceredigion, county of Aberystwyth, there are metal mines which have polluted the waters [7], specifically the Rheidol. Acidobacteria has been observed in mines, soils, and metal-contaminated soils (see section 4.1 on page 20); which creates the first relation of finding Acidobacteria in the sample: the metal mines infecting the waters have potentially contaminated soils: perhaps resulting in them becoming metal-contaminated.

1.2.1 pH

The soil in Aberystwyth is acidic. See figure 1.2 for a representation of soil acidity across mid England and Wales; credit from the online tool: UK Soil Observatory[1] using the *Countryside Survey topsoil pH* (2007) [8, 9].

There are large scale gradients across the UK in terms of soil pH - due to bedrock, primarily. As bedrock is weathered, it forms the mineral bits in soil which then leach out different ions: the concentration of hydrogen ions in the soil increases [10]. Bedrock is dominated by members of Acidobacteria [11].

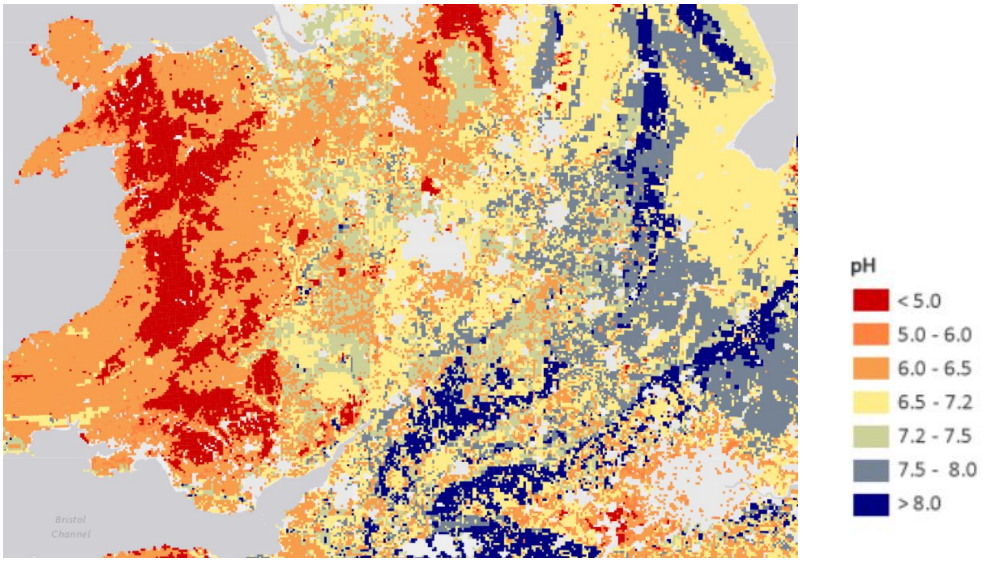


Figure 1.2: Using data from *UK Soil Observatory*, we can observe that Aberystwyth, mid-west coast, soil is more acidic than south-east London - *Countryside Survey topsoil pH* 2007.

The data, from the UK Soil Observatory, inform Aberystwyth having slightly acid 'loaming' soils. pH correlates with with a number of other environmental factors, including

[1] `http://www.ukso.org/maps.html` > Galleries of data: Countryside Survey topsoil maps > Soil pH: Countryside Survey: Topsoil – Soil pH

vegetation and fertility [12]. In 2007, there was a study showing a correlation that the mean pH of soils across the UK was slowly increasing, specifically the low pH areas [13] - these trends have been observed from 1978 to 1998 and 1998 to 2007.

1.2.2 Temperature

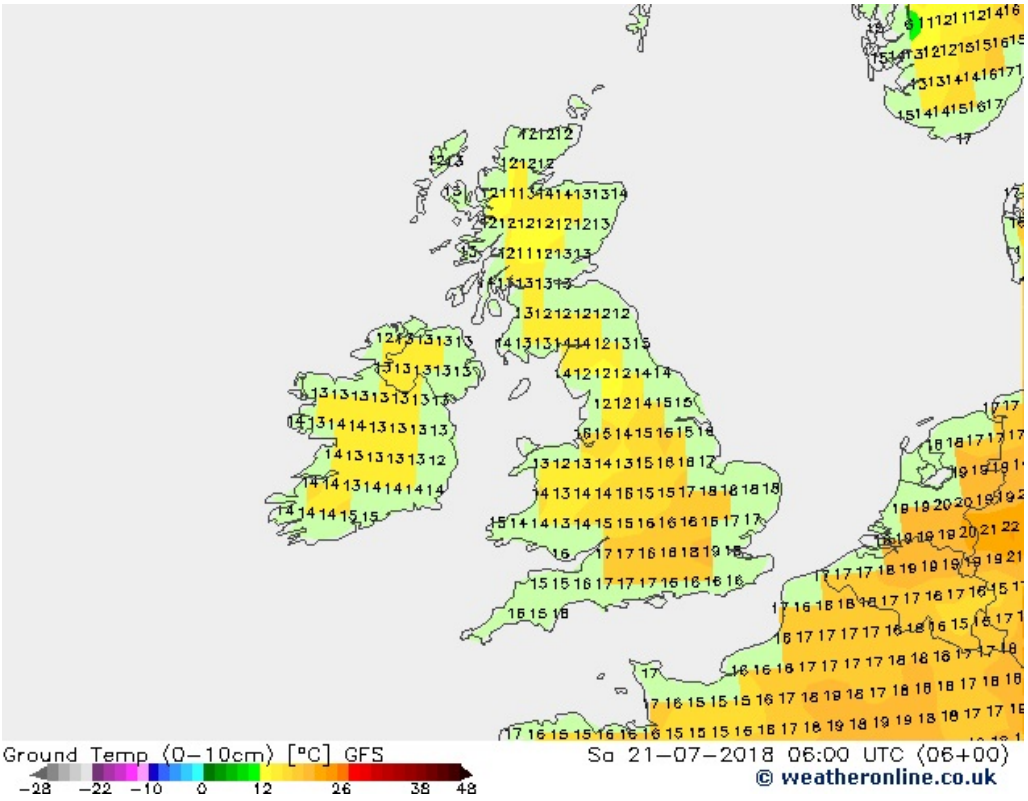


Figure 1.3: Using *WeatherOnline*, we can observe the various soil temperatures across the UK, we would expect the west coast a little lower in temperature than the east due to winds - soil depth: 0-10cm.

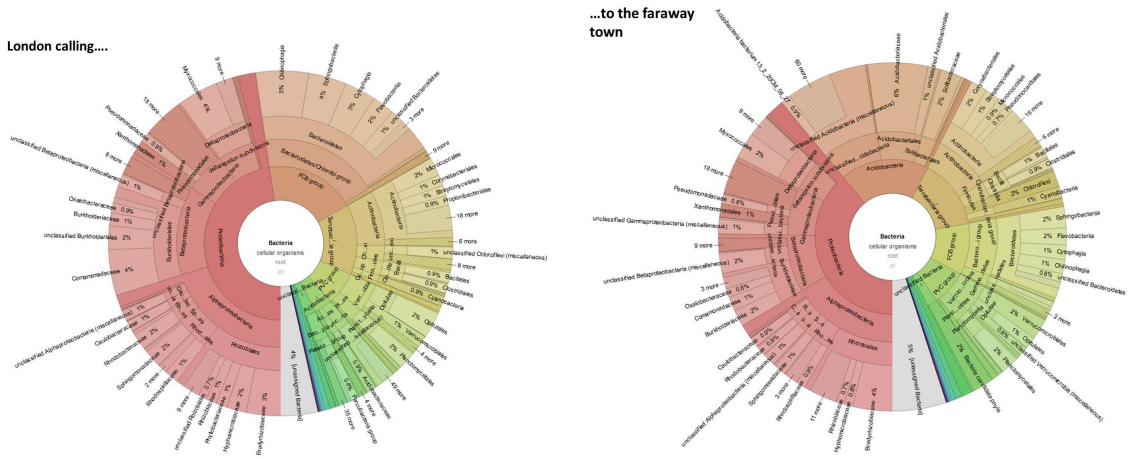
Acidobacteria is said to dominate certain soil temperatures - credit from the online research site: WeatherOnline[2]. From figure 1.3, we can easily conclude that Aberystwyth soil temperature is 14 degrees. Aberystwyth soil is a lower temperature than most inner cities, such as London and Birmingham, however the temperature is higher than most coastal areas on the west. Soil temperature influences the physical, chemical, and microbiological processes that take place in soil [14].

[2] `https://www.weatheronline.co.uk/cgi-bin/expertcharts?LANG=en&MENU=0000000000&CONT=ukuk&MODELL=gfs&MODELLTYP=1&BASE=-&VAR=t1dm&HH=0&ZOOM=0&ARCHIV=0&PRINT=0&PANEL=0&INFO=1&MOUSE=0&MOUSE=1`; Ground Temp (0-10cm) GFS Model

Chapter 2

The Data

BBC Radio 4 were doing a series of interviews in Aberystwyth during 2018. On the 17th of May, 2018, Aberystwyth University researcher and senior lecturer, Arwyn Edwards was involved in the Radio 4 podcast, he sequenced two sets of soil samples and discussed Nanopore (section 2.2) sequencing [15]. Arwyn Edwards, from the research group *Centre for Glaciology*, was assisted by some of his PhD students: André Soares and Aliyah Debbonaire. They conducted the experiment together and afterwards Edwards wrote-up findings plus experience from the event in his blog [16]. To continue, these two soil samples were provided by BBC Radio 4 host, Justin Webb from his garden in London, and the Vice Chancellor of Aberystwyth, Elizabeth Treasure, at Plas Penglais (woodland garden).



(a) London **(b)** Aberystwyth

Figure 2.1: `Krona` plots produced from `Kaiju` of the podcast data-sets. We can observe that there are more Acidobacteria classified sequences in Aberystwyth compared to London. **Note**: colours do not represent the same phylum.

Edwards used `Kaiju` (explained further in section 6.2.2), a program for classifying species from a data-set through a web server. The aim was to generate metagenomic profiles to make some predictions. `Kaiju` provides various plot, figure 2.1 are `Krona` plots (section 6.2.3) which provide some statistical analysis of bacteria diversity plus annotates the species classified from `Kaiju`.

Edwards stated that he found that Webb's *London* compost had marginally more detectable biodiversity: more variety of plant or species in the habitat. He stated that ≈1% of the DNA sequences from Webb's samples were classified as *Propionibacterium (Cutibacterium) acnes*, the relatively slow-growing bacteria linked to the skin condition of acne [17]. This bacterium is common on the skin of adult humans [18] and could potentially indicate contamination of the sample, reflecting the "unorthodox" container (apparently by means of a plastic bag) the compost arrived in. Edwards also noticed from Webb's sample, the presence of *Streptomycetes*, a genus/group of bacteria with over 500 species and have genomes with high `GC` base-pair percentage [19] - it is found predominantly in soil and decaying vegetation.

Edwards flagged that the existence of *Streptomycetes* was also found in the Aberystwyth sample. The Aberystwyth soil was about twice as rich in assignments to the phylum, Acidobacteria, as the London's soil sample. *Acidobacteria* is a group of bacteria that are found in a variety of environments including soil, hot springs, caves, and metal-contaminated soils [20], explained further in section 4.1 on page 20.

2.1 Reads

The experiment Edwards conducted included studying the sample over a series of time stamps. He ran an experiment beforehand in-case there were issues during the event, ran the experiment for the event, and then continued with the study afterwards. I am looking at the data-sets from before and after the event: so no podcast specific event data will be observed. From the podcast experiment, a raw read total of ≈74,000 from Aberystwyth and ≈70,000 from London were obtained - reads are sequences of nucleotides. 6,369 Aberystwyth reads in comparison to 6,461 London reads were classified respectively using `Kaiju`. The read count of the data-sets, which were collected before and after the event, were retrieved from listing 1.

Result: **2,576,848** reads

```
$ wc -l all.fa.fai
```

Listing 1: Word [line] count of a `FASTA` indexed file to observe total number of reads.

2.2 Retrieval

The samples were prepared and sequenced with Nanopore sequencing, a third generation approach used in the sequencing of DNA [21] to output reads. Using the MinION, a portable device from Oxford Technologies[1], the research team used the MinKNOW1 software (older version) with their prepared samples using an improved protocol[2].

[1] `https://nanoporetech.com/products/minion`

[2] One-pot ligation protocol for Oxford Nanopore libraries, Josh Quick *University of Birmingham*, available: `dx.doi.org/10.17504/protocols.io.k9acz2e`

The output reads have corresponding quality scores, explained further in section 2.3. These scores are a measure of quality for the nucleotides generated through sequencing and are on the same numerical scale as phred scores, but are not as accurate - for an example, an average quality score of 10 (phred) for Nanopore is a 1 in 10 error and a score of 20 indicates there will be 1 error for every 100 bases with that score [22]. The scores are common throughout the different sequencing techniques: long-reads like Nanopore and short-reads like Illumina[3]. C. O'Donnel, et al. mentions in their paper that a universal standard needs to be developed for defining accuracy for next-generation sequencing [22]. T. Laver et al. discussed that the error rate with the MinION resulted in difficulties to compare with other sequencing methods - the MinION quality scores do not follow Phred expected error rates, plus possibly has difficulties with high `GC` content sequences [23].

There is lack of information on Nanopore sequencing quality results available - some average quality scores include 8-10. While this could be perceived as low, a score of 8-10 could be deemed as decent quality for Nanopore long-reads. I have experienced this first hand, plus discussed with researchers and read papers [24, 25] that include quality information and plots - note: these papers are pre-prints and have not been peer-reviewed. It seems that there's such an overwhelming amount of short reads with low scores that it 'drags' down others, a method of avoiding this can include filtering out shorter-reads (see section 6.7 on page 54).

R. Lanfear et al. discuss the various reasons for bad quality data: perhaps the extraction of DNA methods plus the MinION runs may 'tire' out [26]. Not all researchers use the same method, have the same equipment, and follow the procedure the exact same. There has also been some research into laboratory contamination affecting microbiome sequencing and analysis [27].

2.3 Format

There are a variety of file formats in bioinformatics and computational biology. For the Nanopore MinION, the raw outputs are `FAST5`. `FAST5` files are used for storing plus managing data[4][5] and some file extensions include `*.f5 *.fast5`

`FASTQ` is another format that can be extracted from `FAST5`. `FASTQ` stores both the reads and corresponding quality scores [28], see listings 2 and 3. File extensions for `FASTQ` includes `*.fq *.fastq`

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Listing 2: An example of a `FASTQ` file format.

[3] `https://emea.illumina.com/techniques/sequencing.html`
[4] Bioinformatics I/O `http://bioinformatics.cvr.ac.uk/blog/exploring-the-fast5-format/`
[5] PoreCamp2016 `https://porecamp.github.io/2016/tutorials/PoreCamp2016-02-MinIONData.pdf`

A list of what each line in a `FASTQ` file coincides with (from listing 2):

❶ Line one: sequence identifier and an optional description (title)

❷ Line two: raw letters of the nucleotides

❸ Line three: optionally followed by the same sequence identifier (and any description)

❹ Line four: quality values (listing 3).

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

Listing 3: A list of quality values that correspond to a sequence; in order from least quality to highest quality.

There is also `FASTA`, again representing nucleotide sequences [29] without the quality annotations (see listing 4). One can gain `FASTA` from the raw `FAST5` or by converting from `FASTQ`, which subsequently results in a smaller file size. There are various file extensions: `*.fna *.fa *.fasta`.

```
>SEQ_ID_ONE
ATGCATGCATGCATGCATGCATGCATGCATGCATGCATGCATGCATGCATGCATGCATGCAT
>SEQ_ID_TWO
ATGCATGCATGCATGCATGCATGC
```

Listing 4: An example of a `FASTA` file format.

`SAMtools`, a set of utilities for interacting with sequences [30], is a tool which proves useful in bioinformatics. One feature of `SAMtools` includes indexing a `FASTA` or `FASTQ`. An indexed file, `FAIDX`, is a much smaller file for fast and efficient access to reference sequences and information within those sequences (see listing 5). The file extension for `FAIDX` usually extends the original `FASTA`/`FASTQ` - e.g. `*.fasta.fai *.fq.fai`.

```
ONE 66  5   30  31
TWO 28  98  14  15
```

Listing 5: An example of `FAIDX` file format (`FASTA` indexed), tab-delimited.

2.4 Relation

For my Masters Dissertation, I won't be studying any of the live broadcast results (podcast results) as Edwards stated they were only a fraction of the data-set. I will only be observing the reads that were produced before and after the event.

Moreover, I will be analysing *only* the data-sets from Aberystwyth; Edwards stated the samples from London were too contaminated. Plus as we are looking at Acidobacteria, the results from Aberystwyth were more rich in this bacteria group.

The data I am using in this project is a merged file of all the `FASTQ` reads (see listing 6), which were converted from the original raw `FAST5` that Edwards provided.

```
$ cat *.fq > all.fq
```

Listing 6: Using `cat` to merge the `FASTQ` files together.

```
$ gzip all.fq
```

Listing 7: Compressing the `FASTQ` file into `gz`.

`all.fq.gz` is the result from listing 7. This `gz` is a compressed version of the `FASTQ` file, which a lot of tools use in order to run faster.

Chapter 3

Project

3.1 Objective

Acidobacteria is one of the least studied bacteria groups as it is fairly new: numerous extracted sequences and species are identified as 'unclassified' due to not knowing which group to be placed in - some are placed in groups however not enough information to be identified properly, see below a brief example of the Acidobacteria tree[1].

- Acidobacteria
  - Acidobacteriia
    - Acidobacteriales
      - Acidobacteriaceae
      - Unclassified Acidobacteriales
    - Unclassified Acidobacteriia
  - Blastocatellia
    - Blastocatellales
      - Blastocatellaceae
      - Pyrinomonadaceae
    - Chloracidobacterium
    - Unclassified Blastocatellia
  - Holophagae
    - Acanthopleuribacterales
    - Holophagales
    - Unclassified Holophagae
  - Solibacteres
    - Solibacterales
      - Bryobacteraceae
      - Solibacteraceae
      - Unclassified Solibacterales
    - Unclassified Solibacteres
  - Unclassified Acidobacteria

[1] `https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=57723`

We are studying Acidobacteria both from a computational biology and bioinformatics perspective. We are using computational methods, such as software, in order to observe the biology, diversity, and species which appear in the data-sets in order to gain an understanding of acidic soils.

Through bioinformatics, the aim is to create a `Python` package in order to take in a `Kaiju` output file of classified, taxaID-labelled reads, and then return the sequences that are specifically Acidobacteria and unclassified Acidobacteria identified taxons. This makes the `Kaiju` output file more informative and useful.

In attempt to make use of some unclassified sequences, we are observing the pattern of `GC` ratio within the various Acidobacteria groups, which these groups have a pattern in being abundant in certain pH soils - explained in more detail in chapter 4 on page 20. There were additional outcomes of the package. The package includes some statistical information: read-lengths, `ACGT` information, and plots. In addition to a `FASTA` output of the unclassified sequences in their groups based on `GC` and pH.

3.2 Chapter Summary

❶ Part I: Introduction

① Introduction: topic and Aberystwyth, Chapter 1
Here I introduce computational biology and bioinformatics and how I will be exploring both. I discuss the town of Aberystwyth: the temperature and pH, all details which will be useful when looking into Acidobacteria.

② Data: type and retrieval, Chapter 2
In this chapter I talk about the data and explain the analysis conducted prior. I introduce Nanopore sequencing and the file formats involved in this project.

③ Project: aims/objective, Chapter 3
In this chapter I discuss the aim/objective of my project.

❷ Part II: Literature Review

④ Acidobacteria: history and background, Chapter 4
In this part of the literature review I'll be presenting Acidobacteria and it's very recent history of only being discovered in the past few decades - the various properties will be discussed and their relevance to the project.

⑤ Related Research: study of methods, Chapter 5
Here I will be discussing the various methods in computational biology that studies sequences and their techniques/algorithms.

❸ Part III: Method

⑥ Investigating the project: looking at the samples, Chapter 6
There will be an established research question and work on the data-set will be conducted in order to get a better look at the samples, e.g. demonstrating tools and methods in bioinformatics.

⑦ Designing & building the package, Chapter 7
This chapter will explain the design and building of the package, what modules are being used and why I chose some methods, plus some basic usage.

⑧ Critical Evaluation: my results, Chapter 8
This section discusses my findings from the running the data-set through my package: presenting the various plots produced and the outputs of my package: what results I find from running the package with the data-set.

❹ Part IV: Conclusion

⑨ Discussion of results, Chapter 9
Talking, plus reflecting, over the collection of results throughout the chapters and how they compare, plus the package results and if they are what we expect compared to other methods.

⑩ Conclusion, Chapter 10
A final overview of the project will be discussed: what went well, what could have gone better, and potential future developments.

Part II

Literature Review

Chapter 4

Acidobacteria

Soils have many organisms that reside, such as bacteria. Bacteria constitute a major portion of diversity in soils [31] and play an essential role in maintaining soil processes [32]. Bacteria is one of many organisms that affects soil formation [33] and chemical exchanges between roots and soil [34].

4.1 Background

Acidobacteria was only recently recognised as a phylum [35] in 2012 [36]. The first species, *acidobacterium capsulatum*, was discovered in 1991 [37]. The rank of phylum is one of the eight major biological classifications taxonomic ranks, ranks ordering hierarchical, see figure 4.1[1]. Acidobacteria is within the kingdom of bacteria and has many classes, which are known as subdivisions.

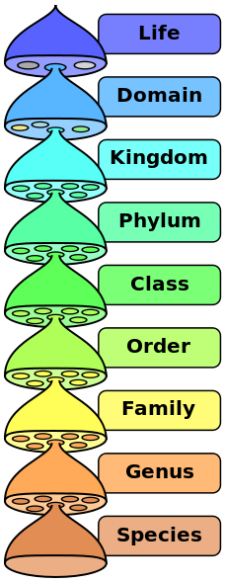


Figure 4.1: Taxon classification ranks. Credit: P. Halasz.

This phylum, Acidobacteria, is one of the most abundant phyla and diverse in Earth soil [38, 39, 40], found in other habitats and environments including soil, hot springs, caves, and metal-contaminated soils [20]. Statistics of Acidobacteria residing up to 52% of the total bacteria community [41].

But our knowledge is quite limited, with research finding that it is similar to Proteobacteria [42]: isolates were originally determined to Proteobacteria [43]. Proteobacteria is another phylum group of bacteria and often observed with Acidobacteria [44], see a phylogenetic tree in figure 4.2 to view the diversity of bacteria. It has be hypothesised that there is a ratio of Proteobacteria to Acidobacteria (P/A), which may provide insight to the nutrient status of soils - low P/A indicating oligotrophic[2] soils [46]; various factors, such as temperature can affect the nutrient-availability in the soil environments [45].

[1] `https://commons.wikimedia.org/wiki/File:Biological_classification_L_Pengo_vflip.svg`
[2] an organism that can live in an environment that offers very low levels of nutrients [45]

It has been predicted that members of Acidobacteria would dominate oligotrophic windswept soils which undergo larger annual temperature variation [47].

As mentioned, the first species was discovered in 1991 [37] and in 2009,the first sequenced genomes of Acidobacteria strains became available. As of 2016 a total of 10 full published are available [48]. From these 10 full genomes, five aspects were brought to attention: carbon use-age, nitrogen assimilation[3], metabolism of iron, antimicrobial, and abundance of transporters.

4.2 Characteristics

The abundance of Acidobacteria is higher in soils with low resource availability (low C mineralisation rate[4]), though this has been disputed: a study finding a positive correlation between the abundance and organic carbon availability [51]. Furthermore, there has been a study of enzymatic activities in Acidobacteria [52]. Some species from a particular genus are able to use lactose, resulting in the possibility of finding this enzyme. Moreover, there has been evidence that some genes in cellular degradation may be involved with the infection of plant cells [53].

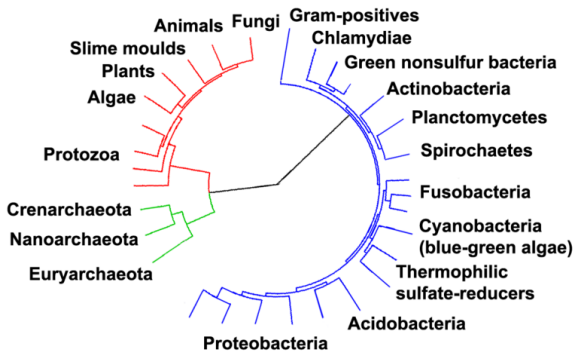


Figure 4.2: Phylogenetic tree showing the diversity of bacteria; colours: archaea green, eukaryotes red, and bacteria blue. Image: public domain.

Despite the diversity, it has been difficult to cultivate [51], poor coverage in bacteria culture collections. Acidobacteria has been observed in mines, soils, and metal-contaminated soils [54], again, possibly explaining, along with the climate, the acidic soils in Aberystwyth due to the metal mines contaminating nearby waters (mentioned in section 1.2 on page 9).

All cultured species have been observed to be heterotrophic[5], some aerobic[6], and anaerobic[7]. There are cases of thermophilic[8] anaerobic species [59], plus acidophilic[9] isolates [61]. Subdivision 1 and 3 (further explanation about subdivisions in section 4.4, table 4.1) species are mesophilic[10] [63]. One specific species, *pyrinomonas methylaliphatogenes* consumes hydrogen H_2.

[3] the absorption and digestion of nutrients by a biological system [49]
[4] decomposition of compounds in organic matter so these nutrients are available for plants [50]
[5] can't produce food, relying on nutrition from sources of organic carbon: plant or animal matter [55]
[6] can survive and grow in an oxygenated environment [56]
[7] organism that don't require oxygen for growth - some react negatively if oxygen is present [57]
[8] thrives at high temperature, e.g. 41 and 122 degree Celsius [58]
[9] thrive under highly acidic conditions (usually at pH 2.0 or below) [60]
[10] an organism that grows best in moderate temperature, neither too hot nor too cold [62]

4.3 pH

Some environment factors drive the characteristics, such as nutrients and pH. pH is said to be the main factor determining the richness and composition of bacteria taxa [47]. As pH increases, bacterial diversity of samples become less variable with low pH soils more diverse [9] - low pH soils have less taxonomic richness.

Acidobacteria is studied to be in bulk quantity in environments with low pH conditions [64, 65]; specifically, acidic soils are dominated by subdivision 1 [9]: class Acidobacteriia. Aberystwyth has high acidity (low pH) due to the climate and possibly the contaminated waters. On a side note, some studies have shown a response to leak roots and abundance in red pepper [66].

High abundance in low pH conditions predicts Acidobacteria is regulated by the soil pH [51], pH correlates with with a number of other environmental factors: including vegetation, fertility [12].

The abundance correlate with pH depends on the subdivisions. For example, subdivisions that appear more frequently in low pH includes: 1, 2, 3, 12, 13 - these subdivisions have a negative relationship with pH. Whilst subdivisions: 4, 6, 7, 10, 11, 16, 17, 18, 22, 25 have a positive relationship [39, 9, 67, 63]. For example: subdivision 1 is highly abundant in low pH (<5) and is less common in soils with a higher pH (>6), whereas subdivision 6 isn't really observed in low pH but becomes highly abundant as the pH increases - showing the evenly distributed taxa [9, 68]. Some soils revealed that grassland soils were dominated by subdivision 6 and forest soils by subdivision 1 [69].

The pH of soils also contributes to the P/A ratio (Proteobacteria to Acidobacteria). Low pH is dominated by Acidobacteria, and when pH increases, Alphaproteobacteria (class of Proteobacteria) became more dominant. Though Alphaproteobacteria is said to dominate Acidobacteria in soils with high nutrient availability [46].

4.4 Project Relation

We have discussed that subdivisions of Acidobacteria are abundant depending on the pH. To express another pattern, the `GC` content of Acidobacteria genomes are consistent with their different phylogenetic placements, e.g. species in the same subdivision (above 60% for group V fragments and roughly 10% lower for group III fragments) are similar, displaying the diversity within the phylum [42]. The difference in `GC` content will also be helpful for the isolation of further genomic fragments of specific subdivisions, either through sequencing or through the construction of `GC`-biased libraries, our project: we can place reads into subdivisions based on `GC` and pH.

There are 10 full genomes available on NCBI Taxonomy[11], however only 9 are available as 1 entry does not have the requested 'ReleaseType', or is suppressed. But in all, there are a total of 157 partial genomes of Acidobacteria as of **Wednesday 18th July 2018** and 159 as of **Tuesday 25th September 2018**.

[11] `https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=57723` > Genome

Below is a list of the full genomes with their corresponding NCBI taxon IDs and showing which subdivision they belong to.

① Subdivision 1

- ➤ Candidatus Koribacter versatilis Ellin345 `NCBI:txid204669`
- ➤ Acidobacterium capsulatum ATCC 51196 `NCBI:txid240015`
- ➤ Granulicella mallensis MP5ACTX8 `NCBI:txid682795`
- ➤ Granulicella tundricola MP5ACTX9 `NCBI:txid1198114`
- ➤ Terriglobus saanensis SP1PR4 `NCBI:txid401053`
- ➤ Terriglobus roseus DSM 18391 `NCBI:txid926566`

③ Subdivision 3

- ➤ Candidatus Solibacter usitatus Ellin6076 `NCBI:txid234267`

④ Subdivision 4

- ➤ Chloracidobacterium thermophilum B `NCBI:txid981222`

⑥ Subdivision 6

- ➤ Luteitalea pratensis (partial strain) `NCBI:txid1855912`

Class	Order	Subdivision	U	Full Genome	Mean	Latest Refseq	Mean	Other	Mean
Acidobacteriia	Acidobacteriales	1		53.28-60.52	58.13	35.18-67.1	57.95		
		2	U					57.6 (NCBI genome)	57.6
Solibacteres		3		61.9	61.9	51.61-73.35	62.17	52.2-53.3 [42]	52.75
Blastocatellia		4		61.24-61.37	61.31	50.5-62.62	58.87		
		5						62.3-68.3 [42]	65.43
		6	U	67.22	67.22				
Holophagae		8				55.14-71.83	66.84		
		10	U						
		13	U					58.5 (NCBI genome)	58.5
		22	U					66.8-67.5 (NCBI genome)	67.15
		23	U					63 (NCBI genome)	63

Table 4.1: Table of the various subdivisions of Acidobacteria and the `GC` range and mean of Acidobacteria subdivisions (annotated classes/orders) from various sources: NCBI full genomes, Latest Refseq, Other (papers and NCBI brief descriptions). The various unclassified subdivisions are noted as U. **Note**: figures rounded up two decimal places.

I used the `GC` section of my package[12] (an older version, which is a basic script) to analyse and provide statistics of all the various Acidobacteria `GC` ratios. I studied the subdivisions `GC` span: from Acidobacteriia (subdivision 1) NCBI[13] groups: downloading the available `FASTA` assembly genomes[14], status: *Latest RefSeq*.

Acidobacteriia (34), Blastocatellia (5), Holophagae (12), Solibacteres (12), Unclassified Acidobacteria (94), and Environmental Samples.

[12] `https://github.com/sap218/python/blob/master/csm9060/acgt-comparison.py`
[13] `https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=57723`
[14] `https://www.ncbi.nlm.nih.gov/assembly/?term=txid204433[Organism:exp]`

Chapter 5

Related Research

Discussing the computational methods to undertake in order to analyse the biology: Acidobacteria content, sequences, and species. Multiple methods have existed for years, with some programs being fairly new. We are mixing the different types of together in order to observe how well they work with this new sequencing method. Despite describing the methods in this chapter, they won't be fully investigated until Part III.

5.1 Grouped Taxonomy

Grouping taxons is a method of understanding the variety of bacteria phylum and species that are residing in a sample. Grouping the data and reads into their phylum groups to observe taxon diversity will overview how much is dominated by which bacterium. We could also investigate this further by linking it with binning (section 5.5) and observe the classes/orders of the various phylum that are present.

5.2 Alignment

Alignment is a method that arranges the sequences in a matrix order to observe regions which are similar: we can extract information from this and see which sequences may have similar functional, structural, and evolutionary relationships [70] - see figure 5.1 for an example of aligning a series of sequences[1]. This method can also be a way of observing which species are present in the sample as sequences have similarities (with a few nucleotides different from the MinION run).

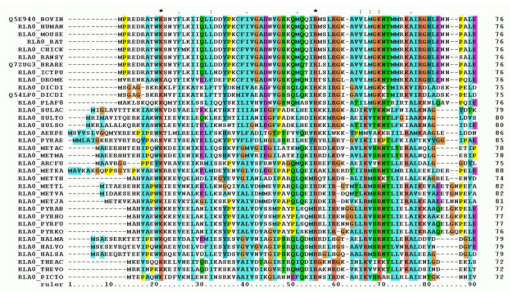


Figure 5.1: Representation of a multiple sequence alignment; credit: *Miguel Andrade*.

[1] `https://commons.wikimedia.org/wiki/File:RPLP0_90_ClustalW_aln.gif`

5.3 Gene Ontology

Gene Ontology *GO* is understanding and representation of the gene attributes across the species being studied from a sample. Some aims of the GO method includes annotating genes and gaining knowledge of this annotation information; plus it provides GO terms which enables a functional interpretation of the sample [71].

5.4 Assembly

Assembly works in order to reconstruct reads into a much longer one finding the sequence similarities. Assembly finds the overlaps in the genes for full genome recovery [6], merging reads can increase the quality of information and one could assemble reads into `contigs` and then with these full assembled genomes could be later used with different methods. See figure 5.2 as an example of assembling short-reads[2].

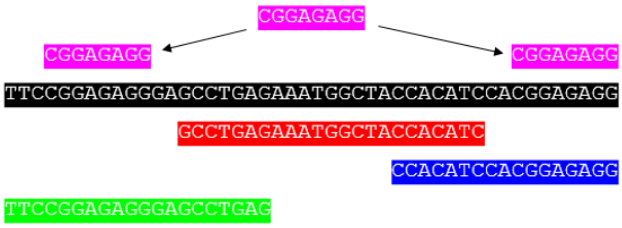


Figure 5.2: Short-read assembly. Image available *Wikipedia*, credit: `Luongdl`.

There are multiple algorithms for assembly - *de novo* recovers areas and segments of a genome that are missed/incorrectly transcribed [72]. There is also, 'reference-based assembly' (co-assembly), which is more memory efficient.

5.5 Binning

Binning is categorising sequences into specific genomes [6]. There are numerous binning methods: 'compositional' binning makes use of the fact that genomes have conserved nucleotide composition (e.g. a certain `GC` or the distribution of k-mers). The output of binning various for each tool/method, some employ self-organising maps *SOMs* or hierarchical clustering. Binning methods are operated in either an unsupervised or supervised (input from user) state to define bins.

[2] `https://commons.wikimedia.org/wiki/File:Seqassemble.png`

Part III

Method

Chapter 6

Investigative Project

Note: see Appendix B for a list of queries and jobs I performed via the Aberystwyth University *IBERS* cluster or entered in a terminal in order to produce the following results.

In this area of the dissertation, the results of a wide variety of tools, programs, and methods will be discussed in order to study the Nanopore sequences and Acidobacteria. This chapter is combing computational biology with bioinformatics: using computational methods and tools to studying the data and creating a solution to our research question (section 7.1).

We are investigating the Acidobacteria content/sequences from the whole sample plus looking at the diversity, in order to do this, I am using the various methods, explained in chapter 5, to gain an understanding of the Acidobacteria coverage and it's species present; through investigating sequence similarity, binning, in addition to observing the species or phylum which lie within the sample. We also want to study Acidobacteria in more detail: for example looking into the GO terms which appear from Acidobacteria reads or from the whole data-set. We would like to try assembling reads to assist in genome building.

As Nanopore is fairly new, we are going to use both common, older techniques (`BLAST`) and newly designed methods (`BusyBee`) for these long-reads. **Note** for future reference: the MinION has 2408 pores but only receive signals from 512 at a time so after a while it changes the pores it is processing (≈8 hrs), these are '**muxes**'.

6.1 Tools

I will be researching the related tools designed for long-read, specifically Nanopore sequences. I am observing how these tools compare to each other and studying the results. I am also looking into products that may be similar to my package idea: discussing their relevance and how I could build my product based on this related knowledge. A variety of the following tools mentioned were created with various programming languages such as: `Python` [73] and `R` Statistical Language [74].

6.1.1 NCBI Taxonomy Tree

`NCBI_taxonomy_tree` is a tool for in-memory mapping. The NCBI Taxonomy database is a curated set of names and classifications for all of the organisms that are represented in *GenBank*[1] in the form of the 2 files: names of the different nodes `names.dmp` and structure of the tree `nodes.dmp`. This tool makes an in-memory mapping of the NCBI taxonomy with a `Python` v2.7 class that maps the `dmp` files in a `dictionary` which can be used to retrieve lineages, descendants, and more. This method was originally an idea I had planned as the Acidobacteria genome retrieval aspect could be related, however, calling the specific names of Acidobacteria would result in missing data: specific named genomes.

As of 03/09/2018, there was no available paper - this tool is available on GitHub[2].

6.1.2 pauvre

`pauvre` is a plotting script designed for long-reads: Nanopore and PacBio (another long-read sequencing developed by *Pacific Biosciences*[3]). `pauvre` was one of the first tools I used in order to observe the read-length and quality of my data. The plot it produced was very useful in order to view the dispersion of the mean quality (Phred).

As of 03/09/2018, there was no available paper - `pauvre` is available on GitHub[4].

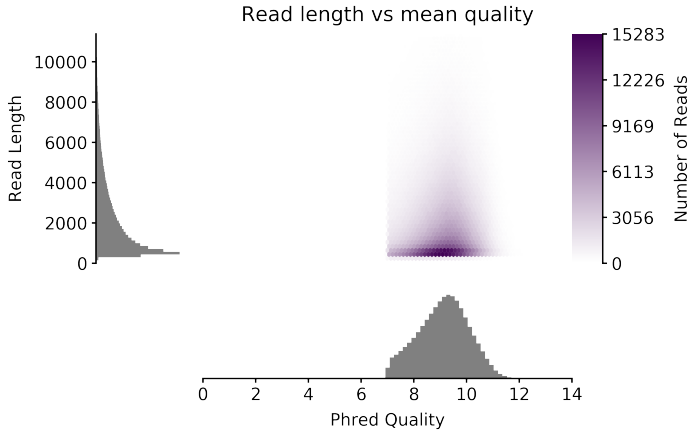


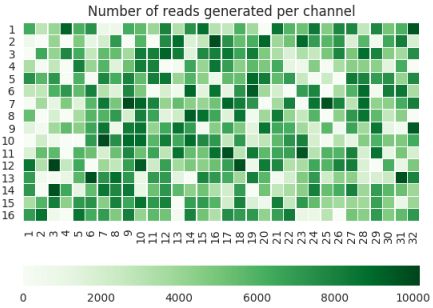
Figure 6.1: Read-length against mean Phred quality heatmap produced by `pauvre marginplot` - displaying the disperse quality pixels with majority collated at Q9 and displaying the dominating short-reads.

Figure 6.1 heatmap quality ranges from ≈7 to <12, with an average of ≈9; short-reads were in higher quantity but limits at 10,000. The statistical output, however, gave more information that the read-length was as high as ≈<60,000 - the read-lengths higher than 10,000 may have been ignored due to low quantity. This tool inspired me to explore read-length in more detail: we can see similarities in `NanoPlot` (subsection 6.1.3).

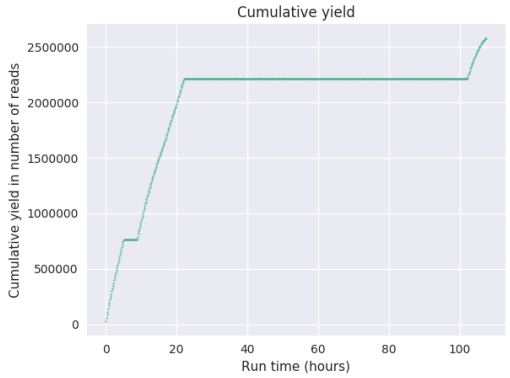
[1] `ftp://ftp.ncbi.nih.gov/pub/taxonomy/`
[2] `https://github.com/frallain/NCBI_taxonomy_tree`
[3] `https://www.pacb.com`
[4] `https://github.com/conchoecia/pauvre`

6.1.3 NanoPlot

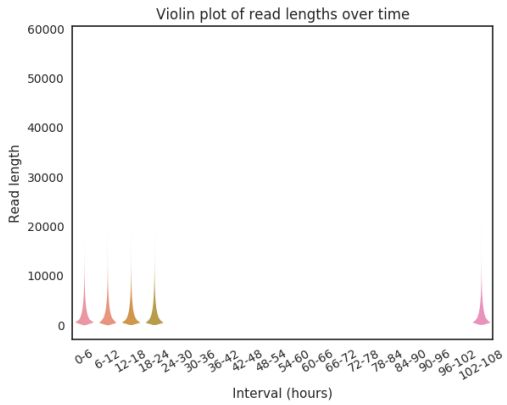
`NanoPlot` is a plotting tool for long read sequencing data and alignments [75] and is available on GitHub[5]. For analysis of the MinION data in further detail (after `pauvre`), I originally planned on using `poretools` (v0.6.0), a toolkit for working with Nanopore sequencing data from Oxford Nanopore [76], however it is outdated and one of it's creators, Nick Loman, suggested `NanoPlot`.



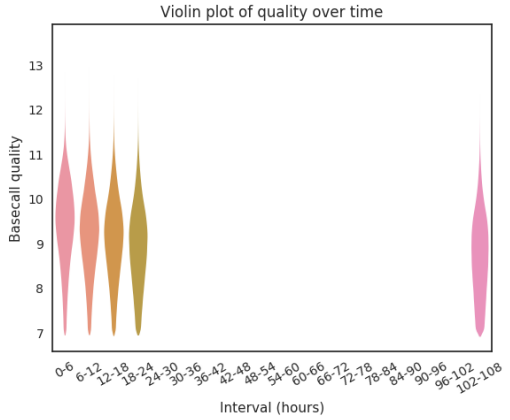
(a) Reads generated per channel (512) of the Nanopore MinION through a heatmap/colour gradient scale.



(b) Time-yield plot showing the number of read produced over time. The flats are time-stamps of a paused MinION.



(c) Violin plot of the read-lengths produced over time, the MinION ran for <30, the gap in plots reflects the time-yield pausing.



(d) Average quality over time, observing the gradual decrease in quality, with the final set much more heavier in low quality.

Figure 6.2: Various `NanoPlot` graphs produced, including: reads per channel, time-yield, and violin plots of read-lengths plus quality over time.

`NanoPlot` is used for analysing the output reads from the MinION by plotting the channel reads, read-length and quality over time-yield - see figure 6.2. We can observe there is a lot of white space for the sub-figures 6.2c & 6.2d plus 6.2b due to the run time of the sequencing being paused - even though the max read-length reaches as high as ≈60,000, there is still a lot of white space making it difficult to gain much information from the graphs - I believe the podcast event data would have completed the gaps here.

[5] `https://github.com/wdecoster/NanoPlot`

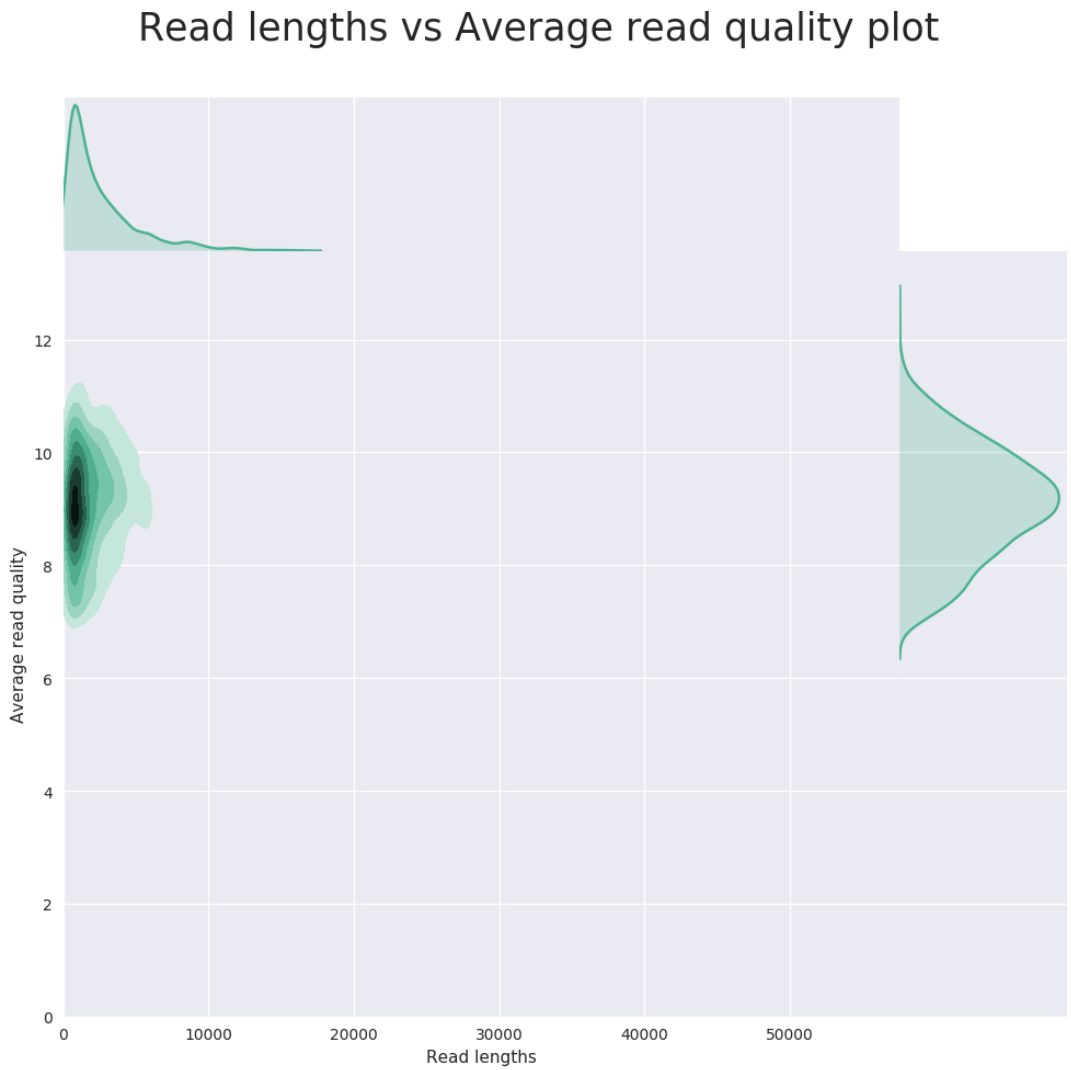


Figure 6.3: Read-lengths against the average quality of read heatmap plot produced by `NanoPlot`.

The previous tool, `pauvre` is implemented in `NanoPlot`, which you can observe from figure 6.3: the similarities in their quality/read-length plots. It seems that `pauvre` (figure 6.1) displays the data in a better perspective from having data centred, condensed and based on the bottom. However, `NanoPlot` seems to include all the data, but a downfall includes a lot of white-space: it also seems to ignore numerous high count read-lengths that are, again, perhaps low in quality.

To improve both `NanoPlot` and `pauvre`, the quality scores and read-lengths which seem to be ignored should attempt to be plotted in order to view the quality of the long-reads despite possibly low quality scores. White space should be avoided in order to view the data better.

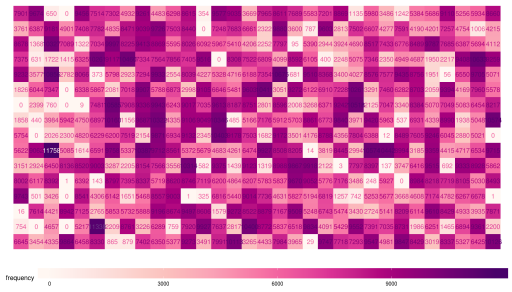
6.1.4 PoreEquality

`PoreEquality` was designed to be ran as part of a Nanopore local basecalling pipeline. From figure 6.4, we can see that `PoreEquality` is very similar to `NanoPlot`, producing channel reads, time-yield, and mean-quality plots.

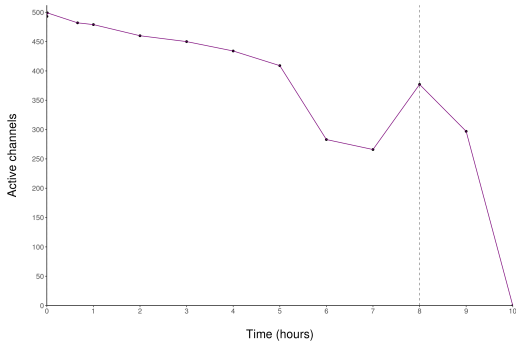
With `PoreEquality`, we can see that the reads generated per channel plot (figure 6.4a) does not have an x/y axis labels (unlike `NanoPlot` figure 6.2a), yet both variations of the channel plots display the data through gradient colour scales, but vertically flipped.

The quality plot seems to cut off at Q9 on the y-axis (sub-figure 6.4d), making it difficult to determine the actual count of reads of quality 9 - we can make a statement that the data-set is Q9 heavy however the lack of visualisation prevents us to know if the data is largely dominated by quality 9 score reads. This seems to be a common issue: lack of axis labels. The read-length histogram (sub-figure 6.4c) also has lack of x/y annotations, despite some available, we won't know the individual read-length count.

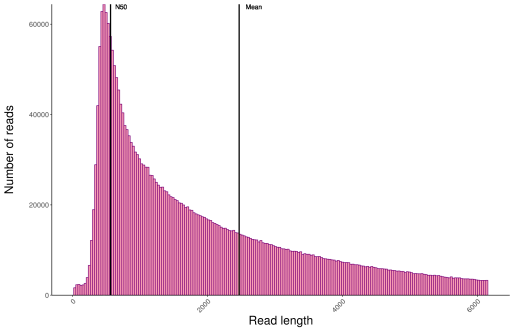
`PoreEquality` also offers different types of plots that look into the channel reads in more detail, we can see the time of active channels in sub-figure 6.4b decreases over time then spikes at 8 hours (mux). However, rather than decreasing steadily it's quite a sudden fall.



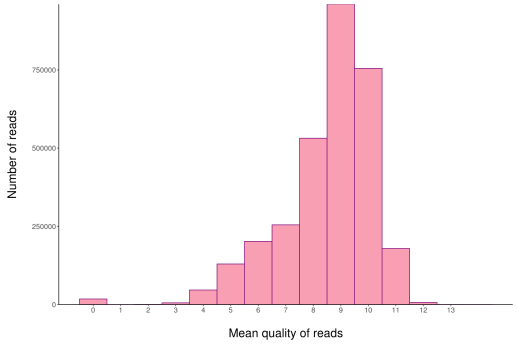
(a) Reads generated per channel of the Nanopore MinION.



(b) Time-yield plot showing the number of active channels.



(c) Histogram of read lengths.



(d) Plot of mean quality of reads.

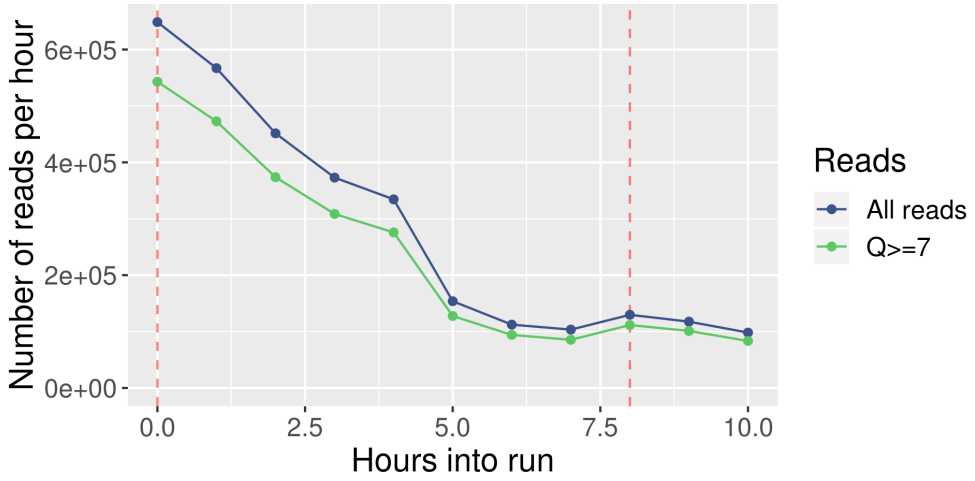
Figure 6.4: Various plots `PoreEquality` produces including reads per channel, time-yield of active channels, plots of read lengths and mean quality of reads.

As of 03/09/2018, there was no available paper but `PoreEquality` is available on GitHub[6].

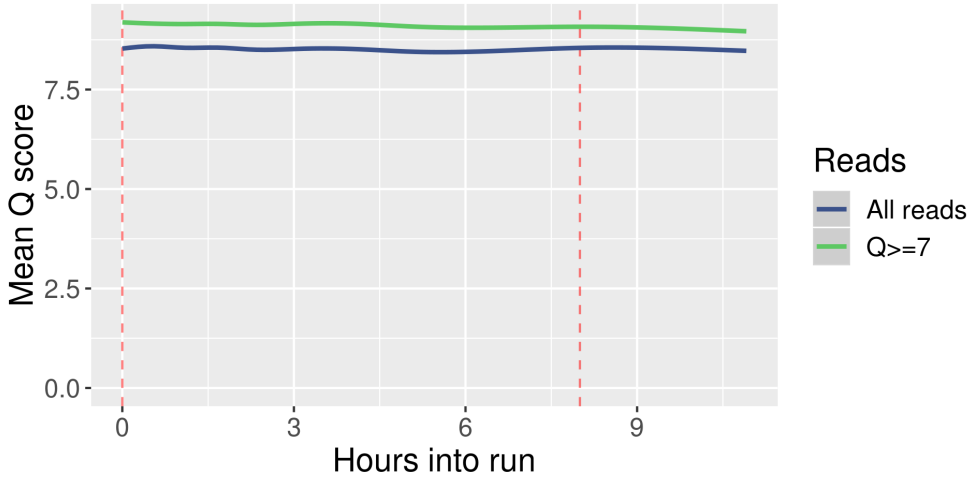
[6] `https://github.com/carsweshau/PoreEquality`

6.1.5 MinIONQC

`MinIONQC` is a tool designed and used as an `RScript` aimed at quality control for MinION sequencing data [26] - script is available on GitHub[7]. `MinION` visualises the muxes in figure 6.5: observing the minor jumps ≈8hrs. Studying the number of reads in sub-figure 6.5a, we can see the count oddly falls suddenly at ≈3.5-5 hrs. Moreover, from the mean-quality plot, sub-figure 6.5b, the scores seems to be difficult to determine: it may average ≈8.5, but we can't know for definite due to the lack of y-axis labels. `MinIONQC` plots seem to label (x&y) integers at half floats, e.g. 2.5, 7.5, and the grid on which they plot each half is 1.25; I believe the axis scales and labels should be consistent with the grid (each full square should be an integer).



(a) The number of reads (y-axis) obtained in each hour (x-axis), each mux results in a noticeable increase in the number of reads per hour.



(b) The mean quality score (y-axis) over time (x-axis).

Figure 6.5: Plots `MinIONQC` produces, including reads per hour and quality by hour. **Note**: muxes, which occur every 8 hours, are shown as red dashed lines.

[7] `https://github.com/roblanf/minion_qc`

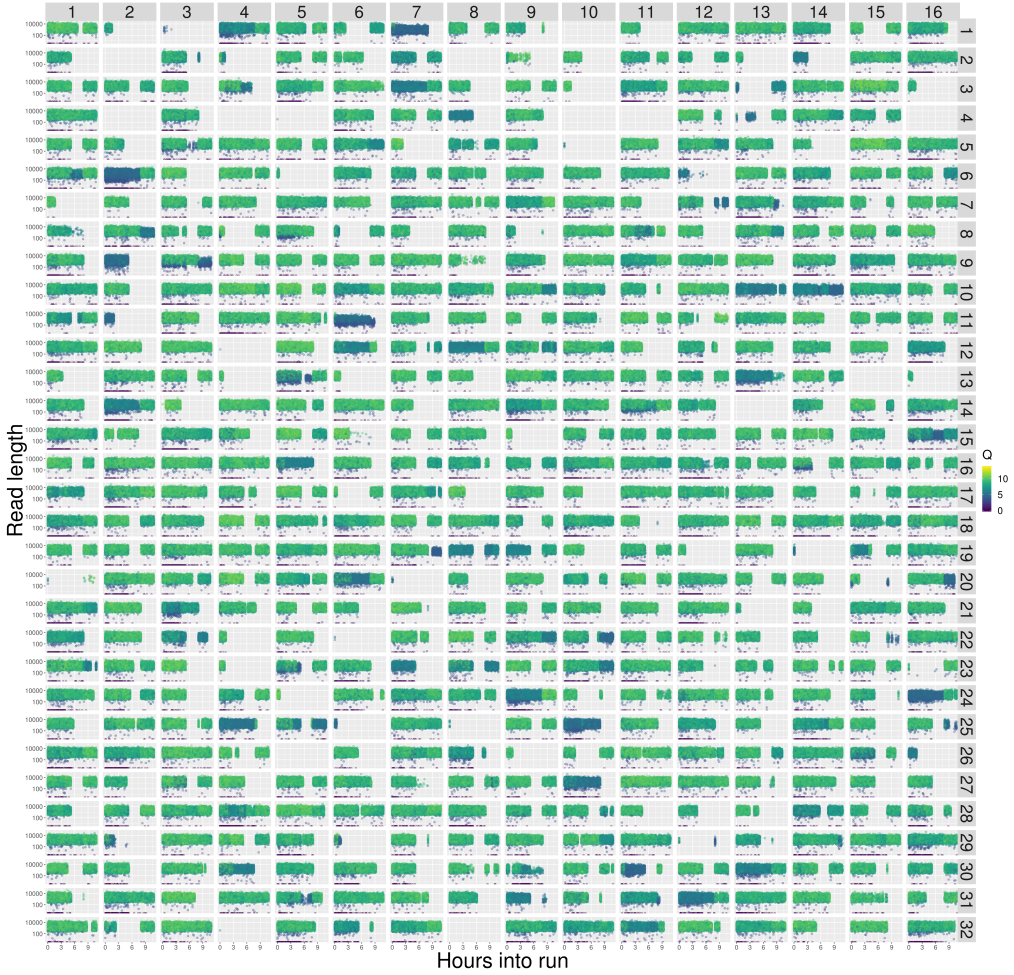


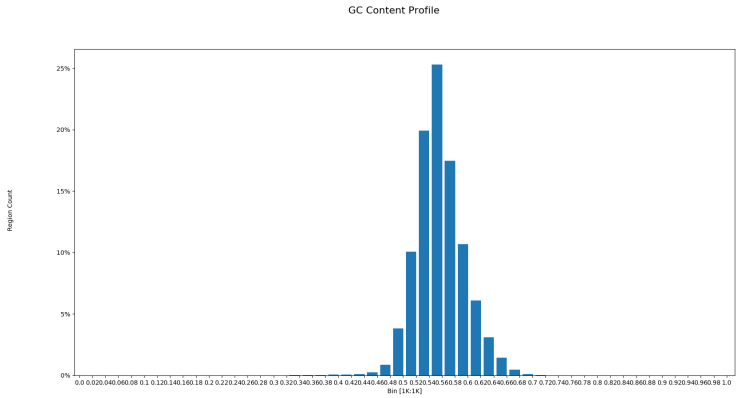
Figure 6.6: Each panel of the `MinIONQC` flowcell plot shows the 512 channels: time on the x-axis, and read length on the y-axis; with points are coloured by the Q score.

Figure 6.6 gives insight into exactly what was going on in each channel over the sequencing process - see frequent (and sometimes extended) periods in which some pores produce only very short, very low quality (blue) reads - `MinIONQC` stating this is due to residual contaminants in DNA extractions blocking the pores (blocked pore are a change in current) - sometimes if a blockage is persistent then one would be able to observe a pattern [26].

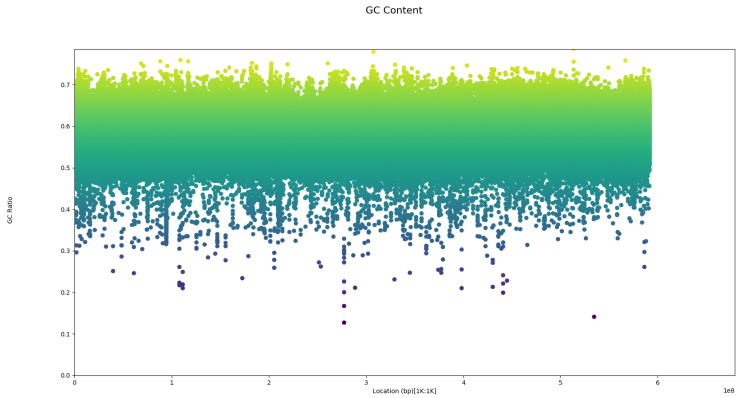
6.1.6 Goldilocks

`Goldilocks` is a tool aimed for locating genetic regions that are "just right" [77]. This `Python` v2.7 class was originally designed for short-reads - available on GitHub[8]. Despite this, I had decided to observe how well it works with long-reads. Some features include plots and statistics for analysing `ACGT` ratio/count with the `NucleotideCounterStrategy` function, figure 6.7. `Goldilocks` is a tool that could be compared to ours as it provides information on `GC` content.

The histogram, sub-figure 6.7a, that `Goldilocks` produces allows for the altering of some bin parameters - though despite entering various numbers, the plot's x-axis seems throughout quite difficult to read (including trying to widen the window size). The scatter plot, sub-figure 6.7b, was designed for a single genome: not a collection of reads - we can somewhat observe the dispersion of the `GC` ratio, ranging densely within the span between 5-7, though it looks as if ≈7.5 is the peak of the `GC` ratio and the `GC` becomes sparse <3: backing up some observations made from my package (see section 7.3 on page 56).



(a) `GC` histogram.



(b) `GC` scatter.

Figure 6.7: `Goldilocks`' `GCRatioStrategy` function creating both histograms and scatter plots in order to observe the `GC` content/ratio along the base-pair location (x-axis).

[8] `https://github.com/SamStudio8/goldilocks`

6.2 Grouped Taxonomy

6.2.1 Kraken 2

`Kraken` v2 is an updated version of `Kraken` that assigns taxonomic labels to short-reads [78]. Despite updated, `Kraken` v2 requires a lot from a system, for example to create a database for `Kraken` it requires ≈100GB of disk space[9]. We struggled to use this method as the database-build job took days until it stopped with an error, occurring multiple times due to lack of computational storage space on the *IBERS* cluster server. Luckily taxonomy labels and classifications can be done by `Kaiju`, which also does it via their cloud server efficiently: the collection of all `FASTA` reads were classified within that same evening. `Kraken` v2 is available on GitHub[10].

6.2.2 Kaiju

`Kaiju` is a program for sensitive taxonomic classification of sequenced reads [79]. `Kaiju` uses the *Burrows–Wheeler transform* for it's matching method: rearranging a character string into runs of similar characters [79]. It offers two separate run modes: *Greedy* (allowing mismatches of classifications) and *MEM* (maximum exact matches). Edwards used it previously on the podcast data to gain taxonomic labels (`Krona` plots figure 2.1). For figure 6.8, using the default *Greedy* values, we can observe a wide phylum diversity: Acidobacteria, Proteobacteria, plus other Bacteria. The multiple bubbles of Acidobacteria represents the various classes, orders, or unclassified: the largest Acidobacteria.

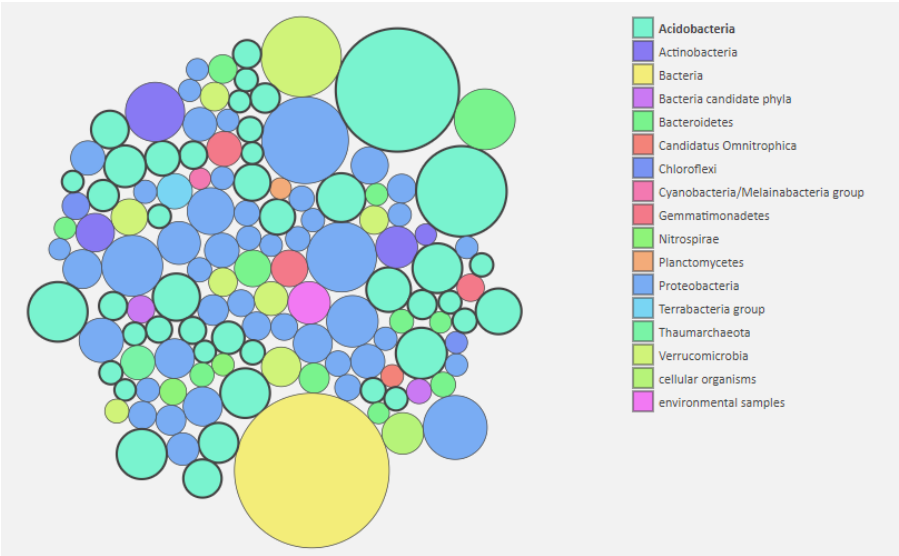


Figure 6.8: `Kaiju` bubble plot with Acidobacteria highlighted.

`Kaiju` database date: 2017-05-16 via the online web-server[11]. `Kaiju` is also available on GitHub[12] for a command line interface *CLI*.

[9] `http://ccb.jhu.edu/software/kraken/MANUAL.html`
[10] `https://github.com/DerrickWood/kraken2`
[11] `http://kaiju.binf.ku.dk/server`
[12] `https://github.com/bioinformatics-centre/kaiju`

1	2	3	4	5	6	7
U	7310a9b5-a04e-4565-a7ce-651bed03d21d	0				
C	f86f4d4a-7735-43eb-a801-0649cd2cc69b	1803434	77	1803434	OLE10643.1	SKLEAAARKEATIQNAEA
U	215fce23-eba8-4993-80f4-9414a69d1897	0				
C	f9e8b1f1-68f8-4463-9eec-6273dd194467	1076588	103	1076588	WP041069373.1	YIQGGGCSGFQYGFEFDENL VYIQGGGCSGFQYGFEFDENL
U	bc87b5d8-1421-4afc-9614-4e76c873a1cd	0				
U	859a50b8-214b-4872-b714-5c2f71ae2f33	0				
U	5426dd91-bde7-4ef6-8bde-f83ad8700c8b	0				
U	25778ee3-7527-4652-9a7a-44d2bd602000	0				
U	a1b0c29b-c13e-492c-89bd-9e16fdf2af49	0				
U	e7563cef-d6e5-4c4d-8a42-414ce528fd58	0				
C	483f913f-1332-43df-abc7-135c4c949453	31989	110	1317123 1443441	WP075220114.1 WP083548865.1	LMGAAYGSAGERCMAVSVAVPVT
U	cbb66560-63b5-4209-8594-fa21bc778cae	0				
U	7c23de16-863d-4b53-b54f-780f3509ab2d	0				
C	ad0e63d3-1084-430d-8119-4c80c7778ab0	186802	77	186802	WP022502618.1	PETARALAVNGAEILFYPT
U	5bebe018-e676-4f71-a8b3-7beff585edb2	0				

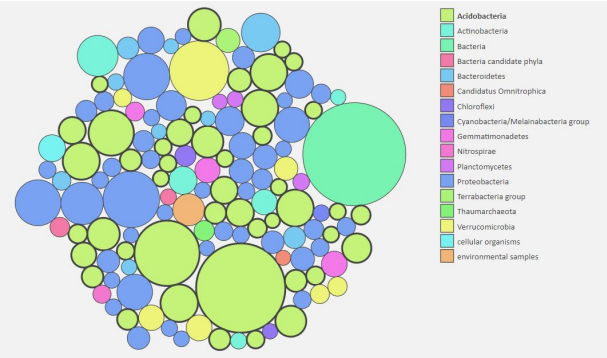
Table 6.1: This table contains the classified and unclassified NCBI taxon identifier. **Note**: some data edited due to LaTeXformatting; wherever specifically 'WP0...' should be 'WP_0...' instead.

Table 6.1 shows a partial (limited to 15 entries) `Kaiju` output file containing one line for each read, 6 tab-separated columns:

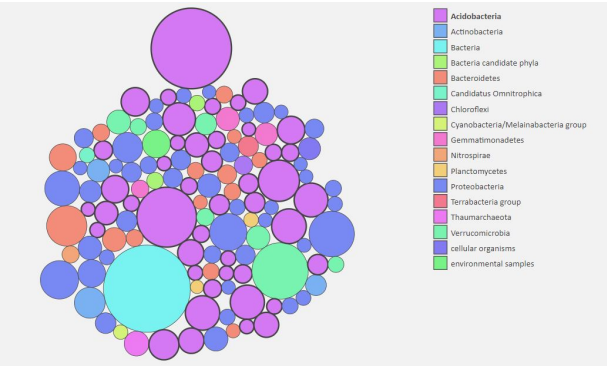
1. either C or U, indicating whether the read was classified or unclassified
2. name of the read/sequence
3. NCBI taxon identifier of the assigned taxon - will be 0 if unclassified
4. the length (*MEM*) or score (*Greedy*) of the best match used for classification
5. the taxon identifiers of best matching database sequence(s), from which the LCA in column 3 is calculated
6. the accession numbers of best matching database sequence(s)
7. best matching database sequence(s)

From this file we extracted the second and fifth columns in order to use it with our package.

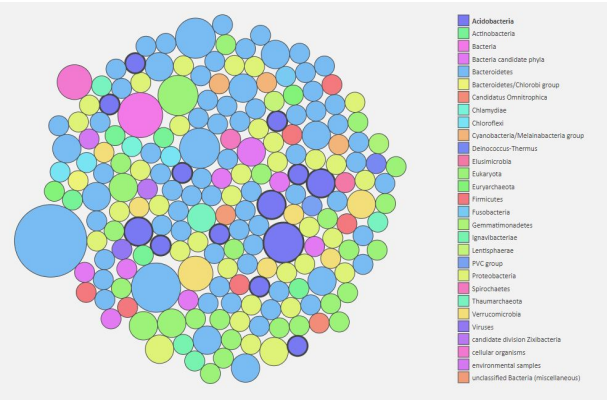
I used `Kaiju` with numerous subsets of the data, see figure 6.9. I created these subsets with `NanoFilt` to filter by quality and read-length (see section 6.7 on page 54). For each subset, it seems that a substantial amount of the data is still dominated by Acidobacteria and Proteobacteria. The quality 12 set, sub-figure 6.9c, was limited by ≈700 reads but Acidobacteria continues to be present in the data - though majority 'bacteroidetes'.



(a) Subset 1: Q9, RL 5000, ≈200,000 seq



(b) Subset 2: Q10, RL 2500, ≈170,000 seq



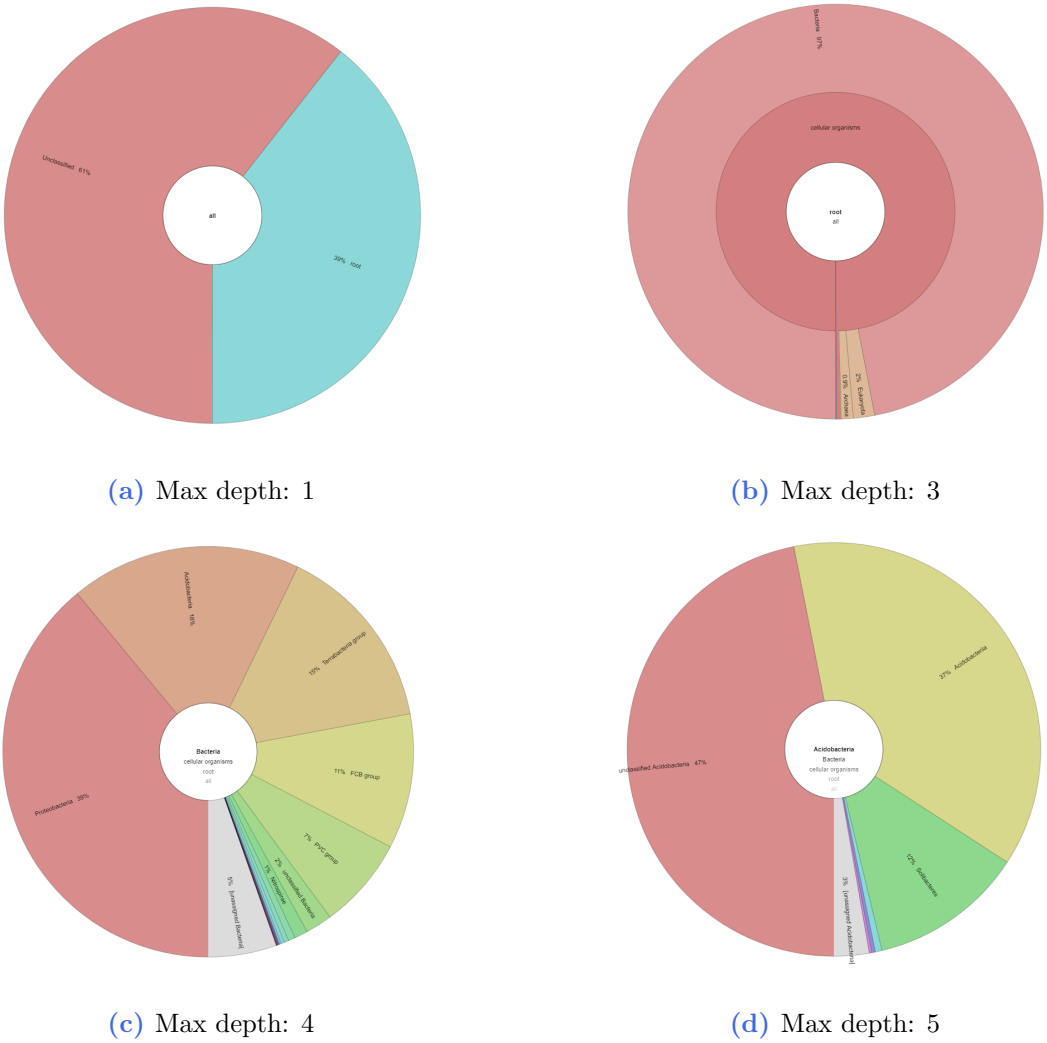
(c) Subset 3: Q12, ≈700 seq

Figure 6.9: `Kaiju` bubble plots of the subset/filtered data ranging from quality 9, 10, and 12 and various minimum read-lengths; Acidobacteria classified taxons are highlighted. **Note**: Q = quality, RL = read-length, and seq = sequences.

6.2.3 Krona

`Kaiju` uses `Krona`, as an extension to visualise the classified taxons: phylum, class, order, and more in-depth taxonomy ranks. `Krona` is an interactive plot for exploration of hierarchical data [80], they represent statistics of the diversity, see figure 6.2.2.

From whole sample, 61% sequences were unclassified, 39% were classified as cellular organisms, see in sub-figure 6.10a. Sub-figure 6.10b shows that of the classified cellular organisms: 97% bacteria, 2% eukaryota, 0.9% archaea. In bacteria, 38% was identified as Proteobacteria, 18% was Acidobacteria - sub-figure 6.10c. Finally, sub-figure 6.10d shows 47% of Acidobacteria classified reads are unclassified Acidobacteria - from the class ranks: 37% Acidobacteriia (15% unclassified), 12% Solibacteres (0.7% unclassified). Acidobacteria was 7% of the whole sample



(a) Max depth: 1

(b) Max depth: 3

(c) Max depth: 4

(d) Max depth: 5

Figure 6.10: `Krona` plots presented through a circular percentage graph. These plots were interactive through increasing the depth of the plot in view inside the various phylum, class, order, and further in-depth ranks.

Figure 6.11: `Krona` plot of subdivision 1 of the phylum Acidobacteria; max depth: 9.

Figure 6.11 is a display of subdivision 1/class Acidobacteriia, showing the classified reads identified from `Kaiju`. We can see that despite being placed in this class, there are still some unclassified reads.

class	order	family	genus	species	individual
Solibacteres	Solibacterales	Solibacteraceae	Candidatus Solibacter	Candidatus Solibacter usitatus	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Candidatus Koribacter	Candidatus Koribacter versatilis	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae			
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Edaphobacter	Edaphobacter aggregans	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Terracidiphilus	Terracidiphilus gabretensis	
Solibacteres	Solibacterales	Bryobacteraceae	Bryobacter	Bryobacter aggregatus	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Silvibacterium	Silvibacterium bohemicum	
Acidobacteriia	Acidobacteriales				
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Acidobacterium	Acidobacterium ailaaui	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Granulicella	Granulicella mallensis	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Terriglobus	Terriglobus roseus	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Granulicella	Granulicella pectinivorans	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Acidobacterium	Acidobacterium capsulatum	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Bryocella	Bryocella elongata	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Granulicella	Granulicella tundricola	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Terriglobus	Terriglobus saanensis	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Terriglobus		
Blastocatellia	Pyrinomonas	Pyrinomonas methylaliphatogenes			
Blastocatellia	Chloracidobacterium	Chloracidobacterium thermophilum			
Holophagae	Holophagales	Holophagaceae	Geothrix	Geothrix fermentans	
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Terriglobus	Terriglobus sp. TAA 43	
Holophagae	Holophagales	Holophagaceae	Holophaga	Holophaga foetida	
Solibacteres	Solibacterales				
Solibacteres	Solibacterales	Solibacteraceae	Candidatus Solibacter	Candidatus Solibacter usitatus	Candidatus Solibacter usitatus Ellin6076
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Granulicella		
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Acidobacterium		
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Candidatus Koribacter	Candidatus Koribacter versatilis	Candidatus Koribacter versatilis Ellin345
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Granulicella	Granulicella tundricola	Granulicella tundricola MP5ACTX9
Holophagae	Holophagales	Holophagaceae			
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Acidobacterium	Acidobacterium capsulatum	Acidobacterium capsulatum ATCC 51196
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Terriglobus	Terriglobus saanensis	Terriglobus saanensis SP1PR4
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Granulicella	Granulicella mallensis	Granulicella mallensis MP5ACTX8
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Terriglobus	Terriglobus roseus	Terriglobus roseus DSM 18391
Solibacteres	Solibacterales	Solibacteraceae	Candidatus Solibacter	environmental samples	uncultured Solibacter sp.
Blastocatellia	Chloracidobacterium	Chloracidobacterium thermophilum	Chloracidobacterium thermophilum B		
Acidobacteriia	Acidobacteriales	Acidobacteriaceae	Acidobacterium	environmental samples	uncultured Acidobacterium sp.

Table 6.2: This table contains lines corresponding to a node in the taxonomic rank and tree with names for the taxonomic levels of Acidobacteria - class/subdivisions identified in the first column.

Table 6.2 shows a `Kaiju` file provided for the `Krona` diagram: taxon path of the phylum Acidobacteria. This table was filtered to show only the phylum Acidobacteria results.

6.3 Alignment

6.3.1 BLAST

`BLAST` for alignment is used for finding regions of local similarity between sequences, comparing nucleotide sequences to calculate statistical significance [81]. A `BLAST` job can run via their web server[13] or CLI.

Using the *IBERS* cluster server, we ran a full `BLAST` of the 2.5 million reads with 400,000 results received after one month - with limited time, I decided to go with an alternative method in order to `BLAST` the reads. I used `pauvre stats` feature in order to observe the statistical read-length and quality scores in order to filter with `NanoFilt` to `BLAST` a smaller subset or high quality and long-reads, see section 6.7 on page 54.

When filtering by quality 9 & read-length 5000 or quality 10 & read-length 2500, the number of reads ≈200,000 which would have taken two weeks, due to limited time of this project we decided to yet again try alternatives: we found `Blast2Go` (section 6.4.1).

6.4 Gene Ontology

6.4.1 Blast2Go

`Blast2Go` is a platform for analysis of genomic data-sets, looking into protein function prediction [82]. We aim to use this interface for studying GO terms. `Blast2Go` is a software with an interactive GUI[14]. Figure 6.12 shows the work-flow I would follow with `Blast2Go`.

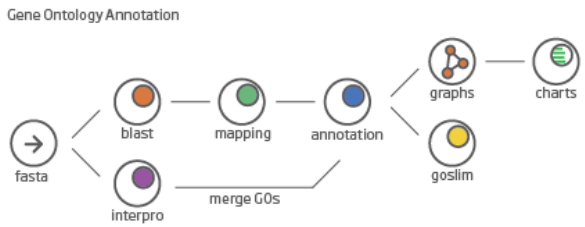


Figure 6.12: `Blast2Go` work-flow for a `FASTA` file.

Using `Blast2Go`, we created a `BLAST` database full of Acidobacteria genomes: a collection of full and partial assembly `FASTA` sequences provided by NCBI: NCBI Taxonomy Browser[15]. With this newly produced database, I ran a local `blastn` job of the quality 10 subset (≈200,000) - see figure 6.13 for the output.

These results seem somewhat what we expect: `Kaiju` stated that Acidobacteria was ≈<16% of the whole data-set, when `BLAST` against only Acidobacteria, the results seem to vary ≈>12%.

[13] `https://blast.ncbi.nlm.nih.gov/Blast.cgi`
[14] `https://www.blast2go.com/`
[15] `https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=57723` > Assembly > Latest RefSeq

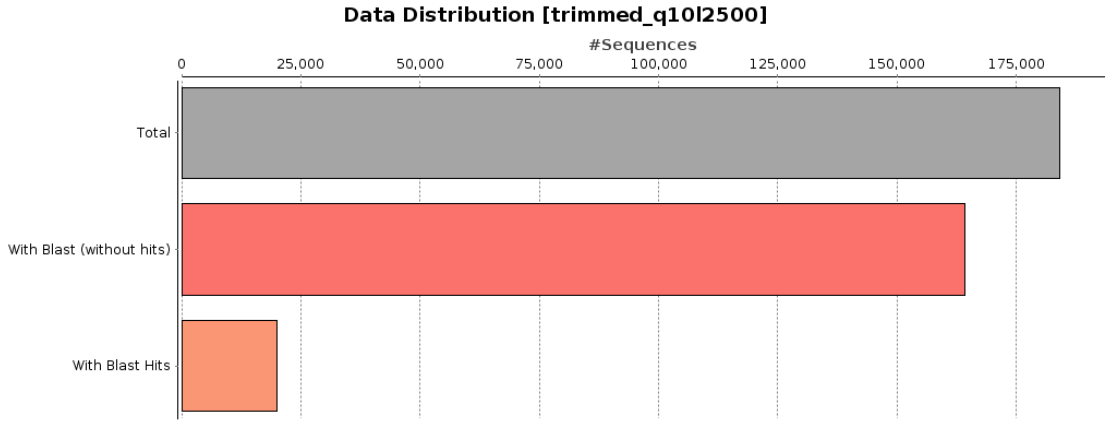


Figure 6.13: `Blast2Go` Q10, RL 2500, ≈200,000 seq `BLAST` results.

Quality 12, Read-length 2500, ≈89 Sequences

Unfortunately my trial of `Blast2Go` **PRO** expired after 7 days and so the `CloudBlast` feature was no longer available to me - I had to make a decision based on the best data-set size and job run for the `Blast2Go` **BASIC** version.

I used `blast-x-fast` on the NCBI server with a subset of the data. `blast-x` uses the `nr` database from NCBI which gains the information of proteins for mapping, annotation and GO terms *goslim* descriptions. One of the first chart it provides includes statistics information: including a read-length histogram (figure 6.14).

Note: see table 6.3 (page 48) for a list of top-5 `BLAST` results with the GO terms annotated.

Note: due to lack of **PRO** account, all images are watermarked with "blast2go".

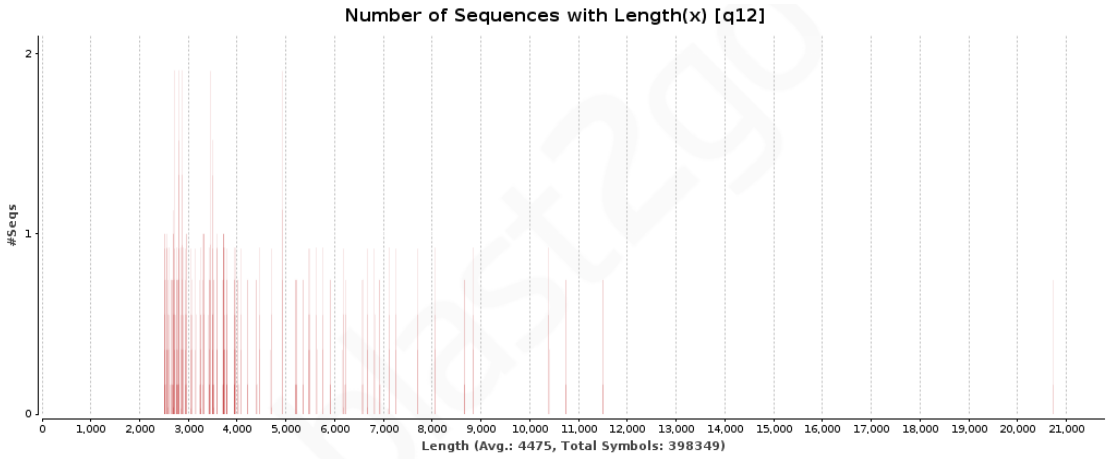


Figure 6.14: `Blast2Go` number of sequences with read-length histogram of data-set: Q12, RL 2500, 89 Seq. We can see the data is limited at 2500 as expected.

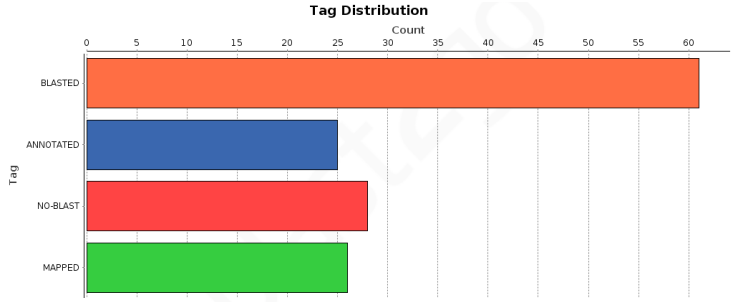
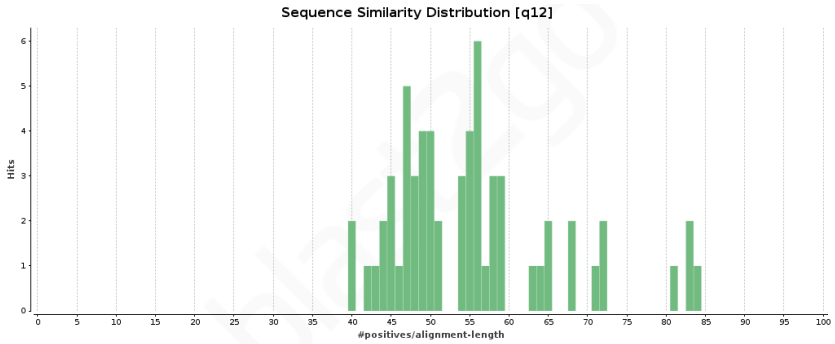


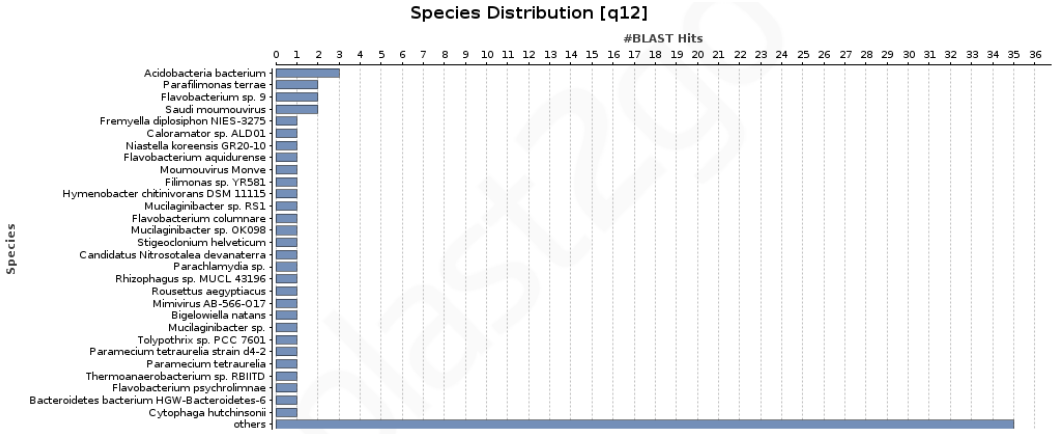
Figure 6.15: `Blast2Go` Q12, RL 2500, 89 Seq `blast-x` summary.

As seen in figure 6.15, out of the 89 sequences, only 61 were successfully `blasted` (28 `no-blast`). 26 were then further `mapped` and then 25 were `annotated`.

Figure 6.16 presents the `BLAST` results: `Blast2Go` provides various charts/plots to visualise the results. From sub-figure 6.16a we can observe the average `Blast` Hit is around 55. Sub-figure 6.16b shows that Acidobacteria is present in this quality 12 sub-set with 3 `Blast` Hits - despite 35 Hits 'others' we can see that Acidobacteria bacterium is the next highest score.



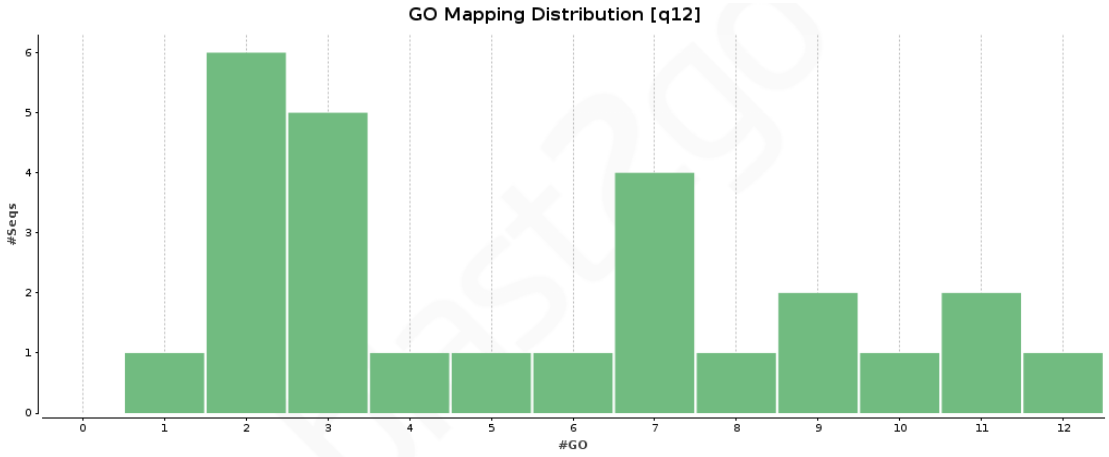
(a) Alignment length/similarity against `BLAST` Hits.



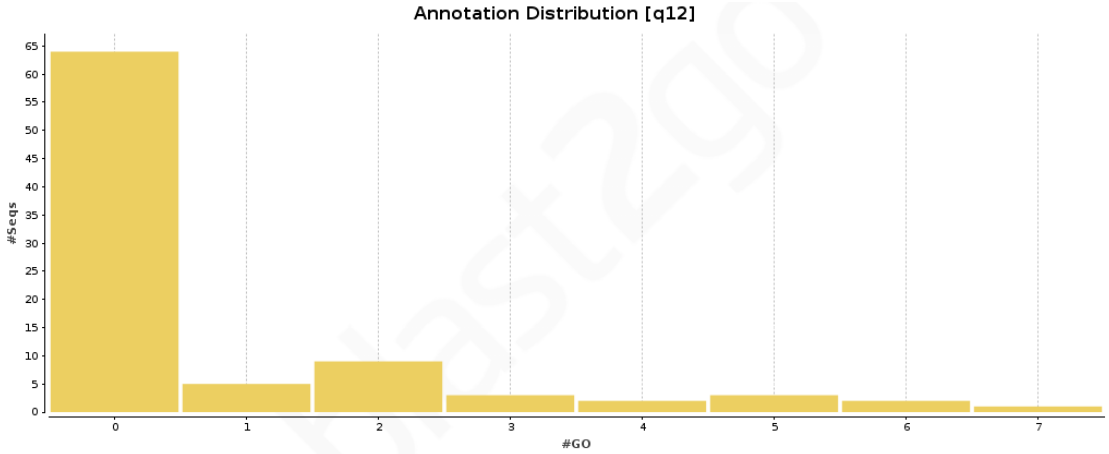
(b) Distribution of species found - with Acidobacteria the second top score with 3 hits.

Figure 6.16: `BLAST` results produced by `Blast2Go` displaying sequence similarity and `BLAST` Hit results.

Figure 6.17 shows the mapping and annotation results in more detail. Of the Q12 data-set, from the 61 (out of 89 sequences) successful `BLAST` results, 26 were mapped (as seen in sub-figure 6.17a) - there was a high of 6 sequences with 2 GO terms mapped, and a high of 12 GO terms mapped to 1 sequence. From sub-figure 6.17b, it shows the whole 89 sequences and the results of GO terms annotated: a count of 25 sequences were annotated - 64 of the 89 sequences had no GO terms, 9 sequences gained 2 GO terms, and 1 sequence gained 7 GO terms.



(a) Count of GO terms mapped



(b) Count of GO terms annotated

Figure 6.17: More `Blast2Go` statistics from the mapping and annotation processes.

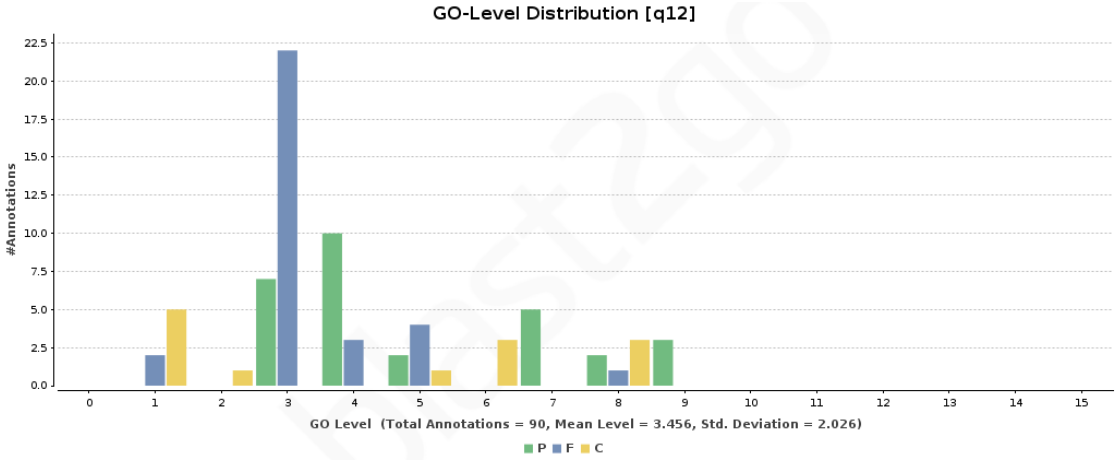


Figure 6.18: GO distribution of the quality 12 sub-set from `Blast2Go`.

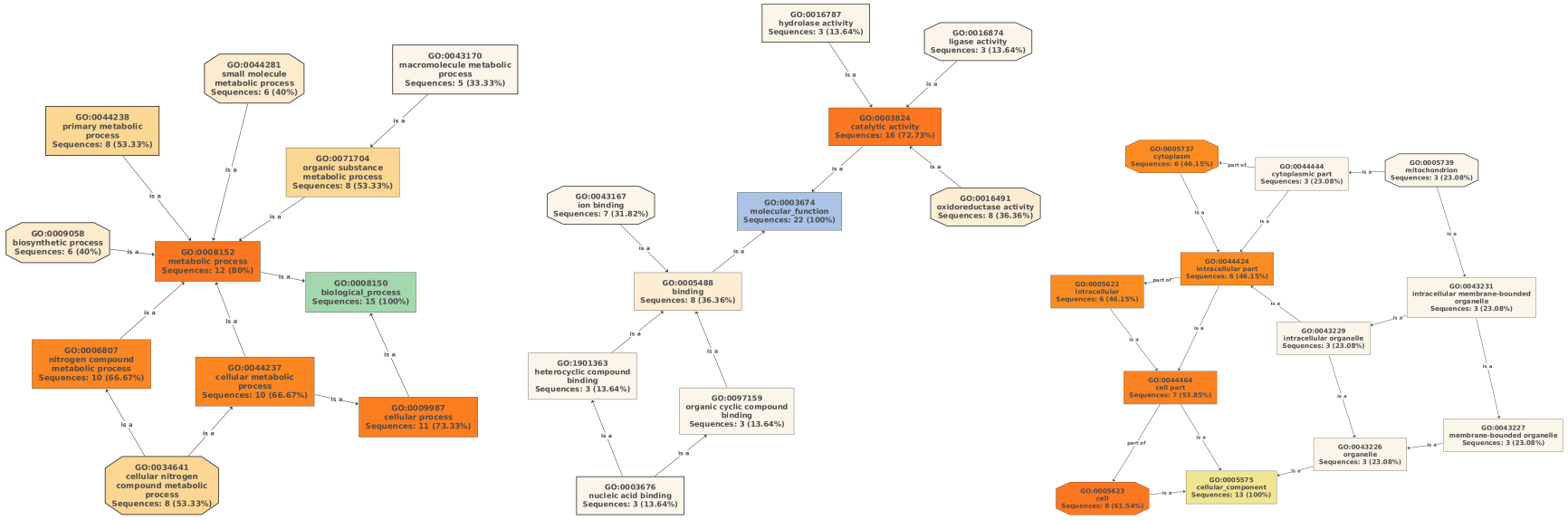
With the limited time a large data-set, I was glad to able to retrieve some GO terms from the smaller Q12 subset (from $\approx 2mil$ reads shortened to 89). There was a great distribution of GO categories obtained, see figure 6.18, unfortunately due to lack of time we didn't have much available to look into the GO annotations in more detail. See table 6.3 for a list of top-5 `BLAST` results with the GO terms annotated and figure 6.19 for organic tree diagrams of the three different types of GO annotations found from the Q12 data. Below are explanations of the three GO terms/categories[16].

P biological processes
A biological process is the objective and outcome of an organism, such as metabolism[17], performed and relied on a set of molecular functions.

F molecular function
A molecular function is carried out by specific gene products, such as catalysis[18].

C cellular component
Cellular components are the parts of a cell, such as structure, compartments, and external environment.

[16] `http://www.informatics.jax.org/vocab/gene_ontology/`
[17] transforming energy by converting chemicals into cellular components
[18] increase in the rate of a chemical reaction by the active site of a protein



(a) **P** - limited to sequence filter of 5 (b) **F** - limited to sequence filter of 3 (c) **C** - limited to sequence filter of 2

Figure 6.19: `Blast2Go` organic graphs of the different GO terms. Pictorial representation is colour co-ordinate by sequence count and limited to sequences to ensure as much data was available to view.

P biological processes, **F** molecular function, and **C** cellular component.

Description	Length	sim mean	GO IDs	GO Names
			P:0006432	P:phenylalanyl-tRNA aminoacylation
phenylalanine–tRNA ligase subunit alpha	6573	84.09	F:0000049 F:0004826 F:0005524 F:0046872 C:0005737	F:tRNA binding F:phenylalanine-tRNA ligase activity F:ATP binding F:metal ion binding C:cytoplasm
hypothetical protein	3150	83.87		
ABC transporter ATP-binding protein	3059	83.33	F:0005524 F:0016887	F:ATP binding F:ATPase activity
CDP-glucose 4 6-dehydratase	5194	81.16		
SRPBCC domain-containing protein	2707	72.86		
			P:0055114	P:oxidation-reduction process
NADH-quinone oxidoreductase subunit D	20730	72.46	F:0048038 F:0050136 F:0051287 C:0005886	F:quinone binding F:NADH dehydrogenase (quinone) activity F:NAD binding C:plasma membrane
			P:0055085	P:transmembrane transport
efflux RND transporter periplasmic adaptor subunit	3788	71.79	F:0005215 C:0016020	F:transporter activity C:membrane
DUF1003 domain-containing protein	2787	68.6		
glucose-1-phosphate adenylyltransferase	2514	68.57	P:0045454 P:0055114 P:0098869	P:cell redox homeostasis P:oxidation-reduction process P:cellular oxidant detoxification
thioredoxin-dependent thiol peroxidase	3715	65.85	F:0004601 C:0005623	F:peroxidase activity C:cell
hypothetical protein SAMD00019534_095250	10394	65.57		
FNIp repeat-containing protein	3713	64.57		
proteasome-associated protein ECM29 homolog	3434	63.41		
			P:0006464 P:0055114	P:cellular protein modification process P:oxidation-reduction process
peptide-methionine (S)-S-oxide reductase	2870	59.57	F:0008113	F:peptide-methionine (S)-S-oxide reductase activity

Table 6.3: `Blast2Go` top-5 BLAST similarity scores (filtered) as an example of some GO results presented.

6.4.2 Diamond

When the premium expired for `Blast2Go`, we tried `Diamond` briefly. `Diamond` is a tool for aligning DNA sequences against a protein (`nr`) reference database [83]. I used `blast-x` to get an thorough search.

Diamond's recent update includes additional features for the `xml` working better for `Blast2Go - xml` is used to store and transport data in a structured file format. We ran `Diamond` on the whole data-set and then used the `xml` with `Blast2Go` but it unfortunately did not work. One possible reason for the lack of result could include an error with the `xml` and `Blast2Go`, or perhaps the incorrect parameters were chosen, but the job on the *IBERS* cluster took too long (plus crashed often due to lack of space) that we didn't have time to continue trying.

6.5 Assembly

6.5.1 Canu

`Canu` was designed for long-read assembly. It has an accuracy of 98% identity reference [84] - available on GitHub[19]. `Canu` was recommended to use, however, the versions of kernel compute nodes on the *IBERS* cluster are out of sync with the head nodes and so are out of date: concluding that `Canu` wouldn't work, we decided to look into other options available.

6.5.2 Miniasm & Minimap2

`Miniasm`, combined with `Minimap2`, is fairly new method of assembly for long-reads, that uses the *de novo* algorithm [85]. Both `Miniasm`[20] and `Minimap2`[21] are available on GitHub.

S. Koren et al. of the `Canu` paper stated that `Miniasm` is a 'magnitude' faster than `Canu` [84]. They also continued stating that `Miniasm` assemblies have both low base accuracy (<90%) and stated to have large errors (average >500bp), with reference at 76.76% identity.

The `Miniasm` results included only 50 full reads with the highest having only ≈16,000bp: short `contigs`. This is due to the diversity of soil, soil has many phylum of bacteria, which all have a variety groups of class, order, and further down the taxonomy ranks; if the sample was something more specific then we would have had better results.

An issue to note is that the `Miniasm` output creates new sequence IDs for the assembled reads, see listing 8, which I would say is a minor drawback: I personally think another output explaining which sequences were assembled would be useful so I can observe the sequences that were assembled together.

```
>utg0000011
CGCGCCAAGGACGGCGCGCGCTCTGGCGGC...
>utg0000021
CGTGTTCATCGTGTCCTCCAAAAACATTTA...
>utg0000201
GCTTCTTCTCCCGCTCGTCGACACTTCTTG...
```

Listing 8: An example of the `Miniasm FASTA` output: it creates new sequence IDs for the newly assembled sequences.

[19] `https://github.com/marbl/canu`
[20] `https://github.com/lh3/miniasm`
[21] `https://github.com/lh3/minimap2`

I used the `Miniasm` output, `GFA`, with a tool, `Bandage`[22] in order to visualise these assembled sequences. `Bandage` is a software tool for presenting *de novo* assemblies through graphical interfaces [86]. We can see the visualisation of the `Miniasm` results in figure 6.20: 50 `contigs`.

Figure 6.20: `Bandage` visualisation of the assembled reads, as we can see there are only 50 assembled sequences but no actual links.

Despite the lack of results, I used the `Miniasm` `GFA` output and converted it into `FASTA` to use it with my package, see listing 33 on page 85. The results included that 15.79% of the whole file were identified as Acidobacteria[23]. I also ran a `BLAST` job on the `FASTA` and some results included max Hit scores of ≈85% however low query cover, below includes the highest Hits:

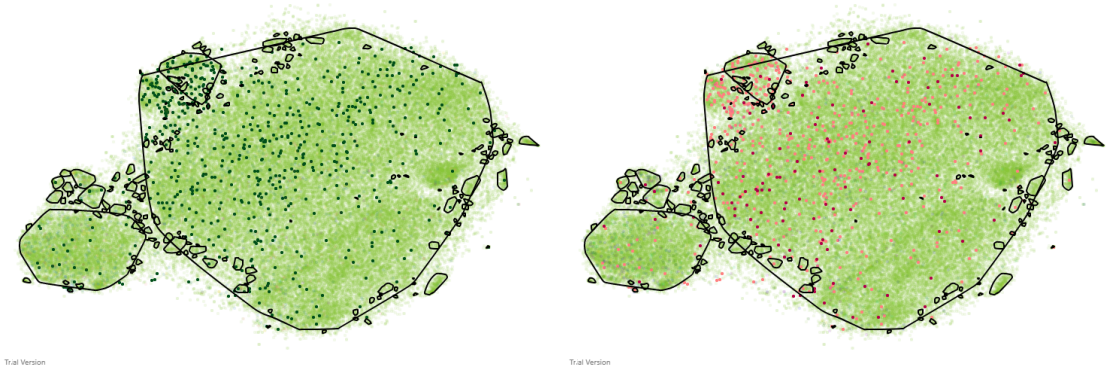
- *Streptomyces scabies* is a bacterium species found in soils globally [87], it interferes with root crops and growth of seedlings. There are other very similar species, such as: *Streptomyces acidiscabies*.
- *Paenibacillus* is an anaerobic (exists free of oxygen) bacterium species [88], coincidentally similar to various Acidobacteria species, from members of subdivision 8 [35], which co-insides with a pH (medium) - Aberystwyth is a pH of around 6.25 (reasons in section 7.4) and so we might find some subdivision 8 species. This species is low in `GC` nucleotide[24] ratios and most resistant to harsh conditions.

[22] `https://rrwick.github.io/Bandage/`
[23] Read-lengths:- min: 8123 & max: 16551
[24] `GC` min of the data-set from the `Miniasm` assembly: 55.047271%

6.6 Binning

6.6.1 BusyBee

`BusyBee` was designed for metagenomic data analysis through a bootstrapped, supervised method [89]. It is available via web-server[25] with current version: v2a7a1ac (2017-08-25). `BusyBee`'s algorithm for it's annotations and binning method is version vfe45da7 (2017-01-09) - it uses `Kraken` (v0.10.5-beta) with `Prokka`[26] (v1.11): "rapid prokaryotic genome annotation".



(a) Phylum taxons, annotated dark green for those Acidobacteria.

(b) Class taxons, annotated pinks for those Acidobacteria.

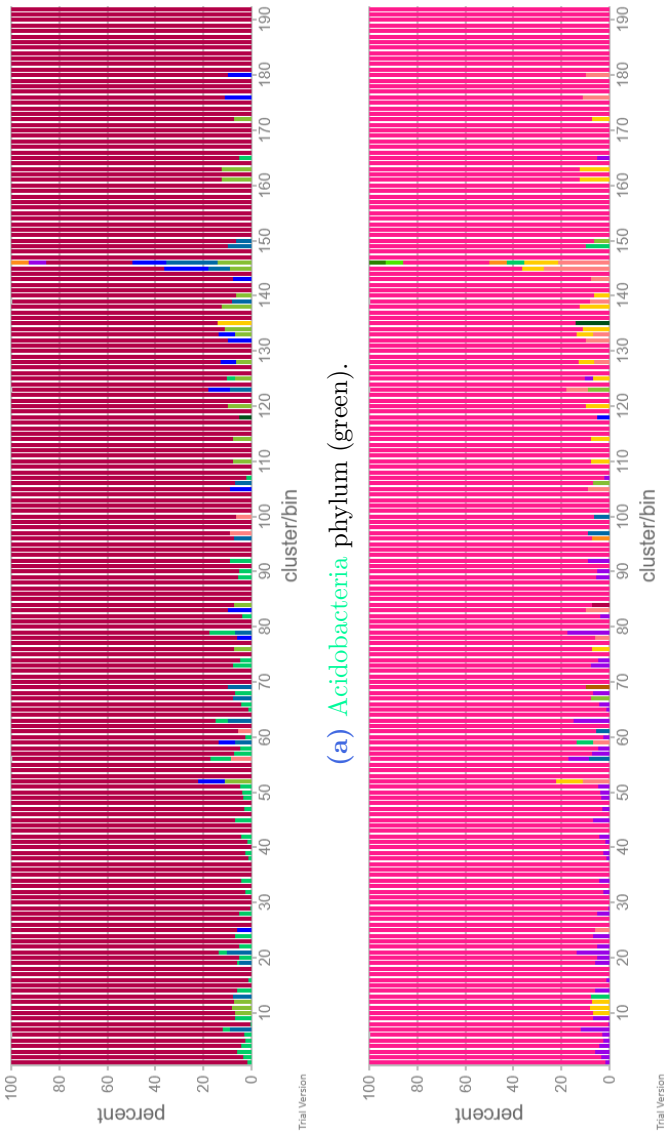
Figure 6.21: Cluster plots produced by `BusyBee`, observing the lack of clusters and Acidobacteria being quite scattered.

Note: Min points in neighbourhood selected = 10.

`BusyBee` was designed for both short and long-reads and aimed for metagenomic data. After reading the paper and observing their cluster plots, I would have expected a variety of clusters with each representing a phylum. However, as we can observe from figure 6.22, there seems to be a lack of clusters: the major/biggest bin is where Acidobacteria seems to reside.

We are unsure why the data does not seem to be place in clusters better. Perhaps this is an issue with the data itself, we had hoped there would be better results compared to the assembly due to the diversity of soil. The data would only run with the Min points in neighbourhood was at 10 - no other run would follow through (errors). Figure 6.22 visualises the cluster/bins via a histogram.

[25] `https://ccb-microbe.cs.uni-saarland.de/busybee/`
[26] `https://github.com/tseemann/prokka`



(a) Acidobacteria phylum (green).

(b) Acidobacteria class (yellow).

Figure 6.22: Cluster/bin graphs produced by `BusyBee`, displaying numerous clusters again with Acidobacteria spread out.

6.7 Assistance

Tools used in order to create the Q9, Q10, and Q12 subsets.

`pauvre`

I used `pauvre` previously in section 6.1.2, but found it's `stats` feature useful for statistical output, see listing 9, of quality/read-length in order to filter.

```
            Number of reads >= bin by mean Phred+Len
minLen       Q6       Q7       Q8       Q9      Q10     Q11  Q12  Q13
     0  2576848  2574028  2212736  1455110   472104   43116  725   10
   250  2574841  2572030  2211603  1454635   472012   43111  725   10
   500  2319796  2317499  2005664  1334169   436273   39276  604    8
   750  1980304  1978435  1719828  1155703   381094   33548  453    5
  1000  1751625  1750041  1524726  1029307   339703   29212  342    3
  1500  1410801  1409602  1231196   834511   274828   22763  211    0
  2000  1152834  1151883  1007848   684091   224200   17929  131    0
  2500   948492   947732   830248   563773   184154   14340   89    0
  3000   782257   781641   685675   466058   151918   11493   60    0
  3500   646117   645629   567084   385864   125519    9307   46    0
  4000   532265   531873   467763   318720   103547    7573   32    0
  4500   438807   438483   386158   263375    85508    6171   27    0
  5000   360899   360641   318072   217199    70378    5016   24    0
  5500   296363   296146   261438   178635    57735    4034   19    0
  6000   243414   243250   214995   146620    47154    3222   16    0
  6500   199365   199239   176312   120126    38477    2544   14    0
  7000   162960   162861   144232    98225    31357    2032   10    0
  7500   132913   132833   117657    80070    25491    1646    8    0
  8000   108103   108032    95729    64981    20719    1317    7    0
  8500    87737    87684    77724    52855    16849    1087    6    0
  9000    71155    71116    63095    42996    13742     876    4    0
  9500    57562    57532    51024    34795    11076     702    4    0
 10000    46456    46432    41198    28152     8939     552    4    0
 11000    30365    30350    26981    18420     5791     340    2    0
 12000    19706    19696    17527    11951     3760     218    1    0
 13000    12778    12772    11385     7755     2395     144    1    0
 14000     8205     8200     7307     5007     1536      92    1    0
 15000     5253     5249     4673     3174      975      63    1    0
 16000     3397     3394     3052     2049      651      43    1    0
 17000     2232     2229     1996     1350      415      28    1    0
 18000     1502     1500     1350      911      291      18    1    0
 19000     1040     1039      946      649      209      14    1    0
 20000      767      766      694      476      159      14    1    0
 21000      558      557      511      354      111      10    0    0
 22000      426      426      391      265       88      10    0    0
 23000      321      321      292      192       63       6    0    0
 24000      250      250      227      149       51       5    0    0
 25000      203      203      183      116       42       5    0    0
 26000      175      175      160      100       38       4    0    0
 27000      150      150      137       86       31       3    0    0
 28000      130      130      120       75       28       3    0    0
 29000      109      109      100       62       23       2    0    0
 30000       96       96       88       53       21       2    0    0
 31000       90       90       82       50       18       2    0    0
 32000       79       79       72       41       15       2    0    0
 33000       73       73       66       37       12       1    0    0
 34000       68       68       61       33       11       1    0    0
 35000       58       58       52       26        8       1    0    0
 36000       51       51       46       23        6       1    0    0
 37000       47       47       42       22        5       1    0    0
 38000       42       42       38       20        4       0    0    0
 39000       38       38       34       17        3       0    0    0
 40000       36       36       32       17        3       0    0    0
 45000       12       12       12        8        2       0    0    0
 50000        3        3        3        2        0       0    0    0
 55000        2        2        2        1        0       0    0    0
```

Listing 9: `pauvre stats` output of the data.

`NanoFilt`

`NanoFilt`, a quality filtering tool [90] was used following the `stats` output from `pauvre`. `NanoFilt` is available on GitHub[27].

[27] `https://github.com/wdecoster/nanofilt`

Chapter 7

Design & Build

7.1 Concept

From studying Acidobacteria from the various research papers, the most interesting idea is that GC ratio is consistent within subdivision groups and these subdivisions are dependent on pH - we wonder if the GC is related to pH? Our research question:

Can we create a Python package that assigns unclassified Acidobacteria sequences into subdivisions based on GC content & pH from the soil sample? Will we observe a pattern of GC, subdivisions, and pH?

A. Quaiser et al. discussed the potential pattern of GC consistency in subdivisions and briefly mentioned the construction of a GC library [42]. I am creating acidoseq in order to observe this consistency further and extract information of subdivisions based on linking GC and pH.

I hypothesise that I can extract unclassified Acidobacteria reads from a data-set and place them into subdivisions based on the GC content and pH. Assigning these unclassified reads into subdivisions based on pH can be somewhat a challenge: there is lack of information of soil samples and their pH, we can only know the pH based on a user's knowledge.

7.2 Software Development

Originally acidoseq was available as a script that user's would run via a command line and the script would request a user input. After further development I was able to successfully create a CLI and make the script into a package and available on PyPI: package index repository for Python.

PyPI: https://pypi.org/project/acidoseq/

GitHub: https://github.com/sap218/acidoseq

7.3 acidoseq

In my dissertation, I am studying a soil sample from Aberystywth and observing the diversity, specifically: Acidobacteria. As stated, Acidobacteria is a fairly new phylum and creating this package allows an in-depth study of it's substantial amount of unclassified reads to gain a better understanding.

I have designed and produced a `Python` package for the analysis of Acidobacteria reads, looking into the `GC` content. `GC` content is consistent in subdivisions - I did sufficient background research to look into this, see table 4.1 on page 24 for the `GC` span across various subdivisions. I aimed to place the unclassified reads of Acidobacteria into their subdivisions based on the `GC` content and the pH of the soil. For example, if the user inputs the soil sample pH of 5 *medium*, there will be `FASTA` outputs of subdivisions 5, 8, and 23 of the reads that have those subdivision certain `GC` span (e.g. `FASTA` of subdivision 5 will have reads that `GC` content lie between 62.3 - 68.3).

`acidoseq` is dependent on a user using **`Kaiju`**. Columns 2 and 5 (see table 6.1 on page 37) are required from a user's `Kaiju` output file; in the Git repository for `acidoseq` (and as seen in listings 30, 31, and 32 on page 84) I provide commands for the user on how to alter the original file to the file that `acidoseq` requires. `Kaiju`'s file of classified reads with taxonIDs is used to link with a list of `NCBI` taxons of Acidobacteria: `acidoseq` provides the NCBI taxonomy of Acidobacteria genomes, both all species and only unclassified species, as inputs - whichever file it uses depends on a user's CLI input.

After linking the `Kaiju` taxonID with the NCBI Acidobacteria IDs, the script outputs the whole set of Acidobacteria reads via extracting the sequence ID (from `Kaiju`) with the original `FASTA`. This output of all Acidobacteria sequences is then further analysed, providing statistics like read-length. However, it focuses on the `ACGT`, specifically `GC`, coverage and subdivision ordering (assuming the user has requested the unclassified reads to be studied). If the use did request the unclassified reads to be analysed, an output of the subdivision-dependent reads is provided based on prediction (e.g. if pH is 4 *low* then unclassified reads are more likely to belong in subdivision 1, 2, 3 and 13 - as explained in section 4.3.

`acidoseq` also has a mapping feature: **`acidomap`**. If a user is unsure of the soil pH, this CLI is recommended to be used first in order to see a visualisation of the soil area. A user enters the city/town via the CLI and the script then plots the location via a translucent circle. The data/colours for the pH plot was obtained from *Countryside Survey: Topsoil – Soil pH*[1] [91] - containing data supplied by *Natural Environment Research Council*. Data of soil pH resource is available under the *Open Government Licence* (OGL) and relevant key publications: *Digital Object Identifiers* - this research study explained that their data/information could be used with sufficient citation/references to owners, which I have met their requirements.

[1] `http://www.ukso.org/Cs/CS_Soil_pH.html`

7.3.1 Code

`acidoseq` is a `Python` v3.5 package - many developers use `Python2` as they are familiar with it, whilst I opted for `Python3` (a newer version) to avoid software/hardware issues. Listing 10 displays the CLI options and parameters, for examples of real use-age see listing 28 on page 83.

```
$ acidoseq --help
Usage: acidoseq [OPTIONS]

Options:
  --taxdumptype TEXT  Study "ALL" or only unclassified "U"?
  --kaijufile TEXT    Place edited Kaiju (csv) in directory for ease.
  --fastapath TEXT    Place FASTA in directory for ease.
  --style TEXT        ['seaborn-bright', 'seaborn-poster', 'seaborn-white',
                      'bmh', 'seaborn-darkgrid', 'seaborn-pastel',
                      'grayscale', '_classic_test', 'ggplot', 'seaborn-
                      whitegrid', 'seaborn-dark', 'seaborn-muted', 'seaborn-
                      colorblind', 'seaborn-ticks', 'Solarize_Light2',
                      'seaborn-notebook', 'dark_background', 'fast',
                      'seaborn', 'fivethirtyeight', 'seaborn-paper', 'seaborn-
                      dark-palette', 'seaborn-talk', 'classic', 'seaborn-
                      deep']
  --plottype TEXT     "span" range of GC means or "line" average mean GC
  --ph TEXT           pH of soil, use map script for assistance.
  --help              Show this message and exit.
```

Listing 10: List of CLI paramters for `acidoseq`.

I used various modules that are included in `Python`, such as `csv`, however others had to be installed through a Linux terminal; such as `matplotlib` (see listing 27 on page 82).

Dependencies:

- `os` - for different operating systems use: e.g. opening and editing files.
- `csv` - for the `Kaiju` output file and the list of Acidobacteria taxonIDs.
- `pysam` [92] - using `FASTA` files easier: opening and gaining the sequences/references.
- `collections` - counting `ACGT` nucleotides.
- `matplotlib (matplotlib.pyplot, matplotlib.patches)` [93] - plotting feature.
- `random` - labelling the annotation text on graphs in order to avoid overlapping.
- `termcolor`[2] - when a user is going through the command line script the out text/stats are highlighted in colour in order to make it user-friendly and more readable.
- `colorama`[3] - for the colours to work with Windows.
- `click`[4] - command line interface that allows a variety of options and default values.

[2] `https://pypi.org/project/termcolor/`
[3] `https://pypi.org/project/colorama/`
[4] `https://pypi.org/project/click/`

7.3.2 Process & Output

Note: see the list of outputs in listing 11, plus further explanation of the output and plots will be included in chapter 8, specifically section 8.1 on page 60.

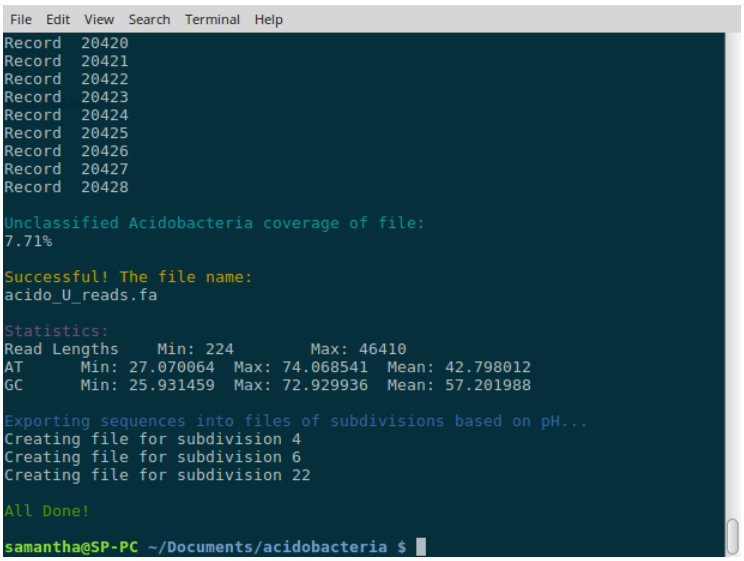


Figure 7.1: Process of `acidoseq` through a Linux terminal.

I believe adding the colours through the script process makes the information easier to read and understand: many CLIs are white text, which personally isn't exactly user-friendly. It highlights all the important information: coverage percentage, the output file name, and others - users will be more inclined to read the coloured text.

Figure 7.1 displays the process of `acidoseq`. First a user would use the CLI to choose the parameters/options, which they wish to run. Once submitted, the script runs each record from the `Kaiju` output file with the NCBI Acidobacteria taxon list. It compares the link count with the amount of reads in the `all.fasta` and provides a coverage percentage. With our data-set, out of the $\approx 2mil$ reads 28,428 were identified as unclassified Acidobacteria taxons =7.71%.

Of all these reads, `acidoseq` makes a `FASTA`, which the script tells the user the name of this collection of sequences - this is where `pysam` was useful: I was able to use their simple feature to gain the sequences rather than making a function to do this. This output file (e.g. `acido_U_reads.fa`) is used via `FAIDX` (`FASTA` index) for easy access and further for statistics of the Acidobacteria reads. The script then uses this file to provide some statistics of the Acidobacteria reads: read-length and the `ACGT` minimum, maximum, and mean figures.

The script then produces `matplotlib` to produce the plots corresponding to these figures: an `AT` to `GC` comparison plot of the base-pairs: ratio comparison with means annotated (`acgt-comparison_unclassified_style-bmh.png`). Plus the other plot of the `GC` means with subdivisions annotated (regions = 'span', and means = 'line'), based on pH (`gc-ratio_unclassified_ph7.84_plot-line_style-bmh.png`).

The final output includes the `FASTA` subdivisions based on pH and `GC`: e.g. `sub4_U_ph7-84.fasta`.

```
acido_U_reads.fa
acgt-comparison_unclassified_style-bmh.png
gc-ratio_unclassified_ph7.84_plot-line_style-bmh.png
sub4_U_ph7-84.fasta
sub6_U_ph7-84.fasta
sub22_U_ph7-84.fasta
```

Listing 11: List of outputs from `acidoseq` when a user requests unclassified reads and inputs a pH of 7.84.

From listing 10, I chose to allow users to add their own style for plots (the script uses a random style if a user doesn't choose one or enters it incorrectly). I wanted to give users as much personal preferences as possible: some individuals prefer the style of `ggplot` while others may prefer `seaborn`.

7.4 acidomap

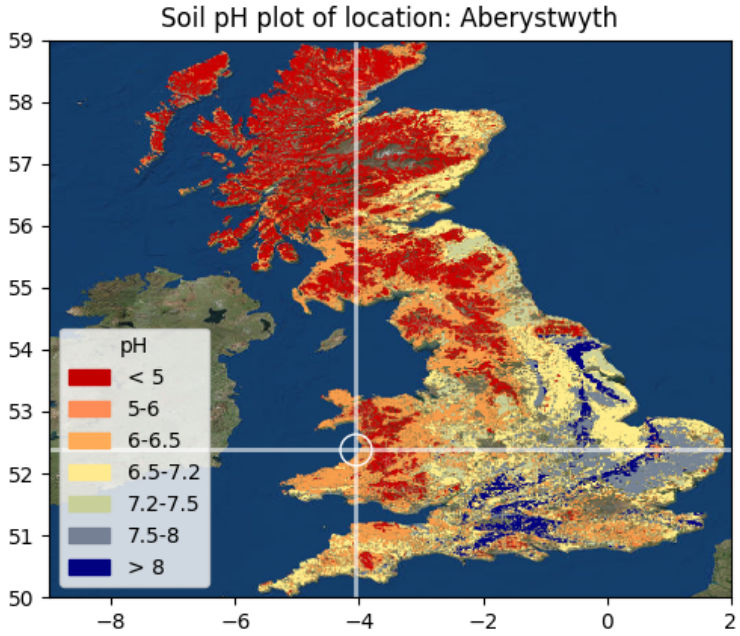


Figure 7.2: Using `acidomap` to gain information of the pH for Aberystwyth. **Note**: due to the fact that the Earth is spherical and maps are 2-dimensional, there will be some distortion when plotting locations.

Figure 7.2 shows the `png` of Aberystywth, see listing 29 (page 83) for the command line query performed. The result could be deemed as 5-6, but I chose 6.25 as the pH due to mean pH of soils in the UK are slowly increasing, specifically the low pH areas [13] - see figure B.1 on page 83 for further evidence why I decided to use 6.25 for the pH.

Chapter 8

Critical Evaluation

I am evaluating my package, `acidoseq` when using the Aberystwyth data-set against it, see listing 28 on page 83 for the command line query performed. Listing 12 shows statistics provided by `acidoseq`. From the data-set, there was a 7.71% coverage: meaning of those $\approx 2mil$ reads, 7.71% were classified as unclassified Acidobacteria from `Kaiju`'s method.

```
Unclassified Acidobacteria coverage of file:
7.71%

Statistics:
Read Lengths        Min: 224        Max: 46410
AT        Min: 27.070064        Max: 74.068541        Mean: 42.798012
GC        Min: 25.931459        Max: 72.929936        Mean: 57.201988
```

Listing 12: Some statistics of the data produced by `acidoseq`.

8.1 Results

From listing 13, by collecting a word-count of the `FAIDX` we can see that from $\approx 2mil$ reads, only 156,676 were classified as unclassified Acidobacteria from `Kaiju` $\approx$7.8%.

```
$ wc -l acido_U_reads.fai
156676 acido_U_reads.fai
```

Listing 13: Word [line] count of the `FAIDX` collection of unclassified Acidobacteria reads.

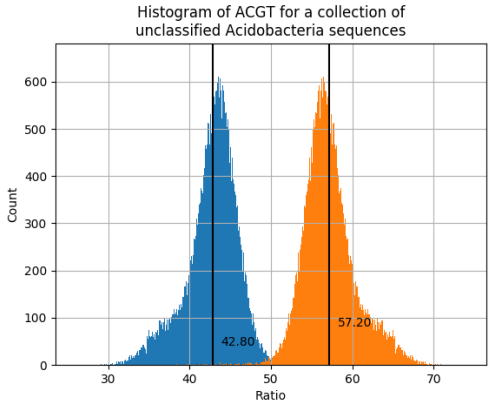
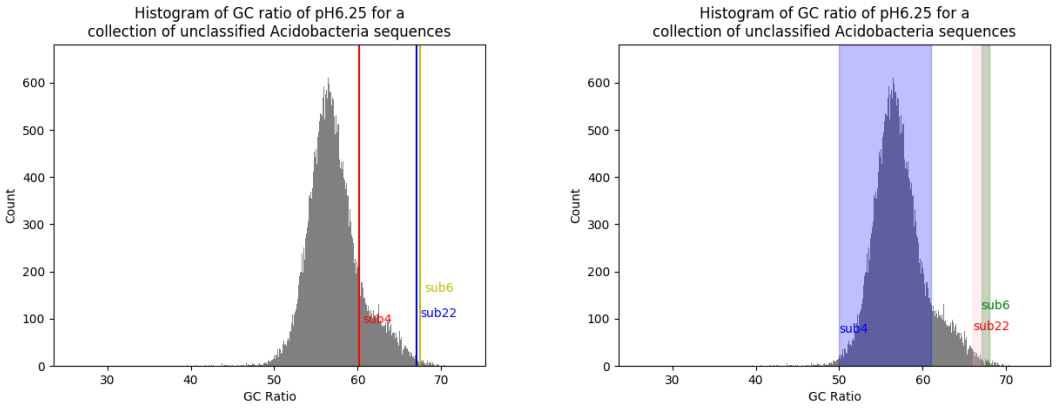


Figure 8.1: `AT` and `GC` comparison plot from `acidoseq`, with means labelled.

Figure 8.1 shows a basic visualisation of the base-pair ratios - I believe this plot would provide information about the reliability of the MinION run. DNA with low `GC` is less stable [94]. Despite the data having some long-reads, it seems to be lower than a lot of other research data-sets, this could be due to lack of equipment or PCR[1] method. From a personal experience, high `AT` can be a result from the MinION sequencing run [24] - I observed repetitive and high count of `AT` and noted these were linked to bad quality data. Moreover research has shown `AT` nucleotides are known to be higher in bias when the sequence is short [96]. I wanted users to compare base-pairs to gain an understanding if their data is not producing results as expected (low quality/unstable data).



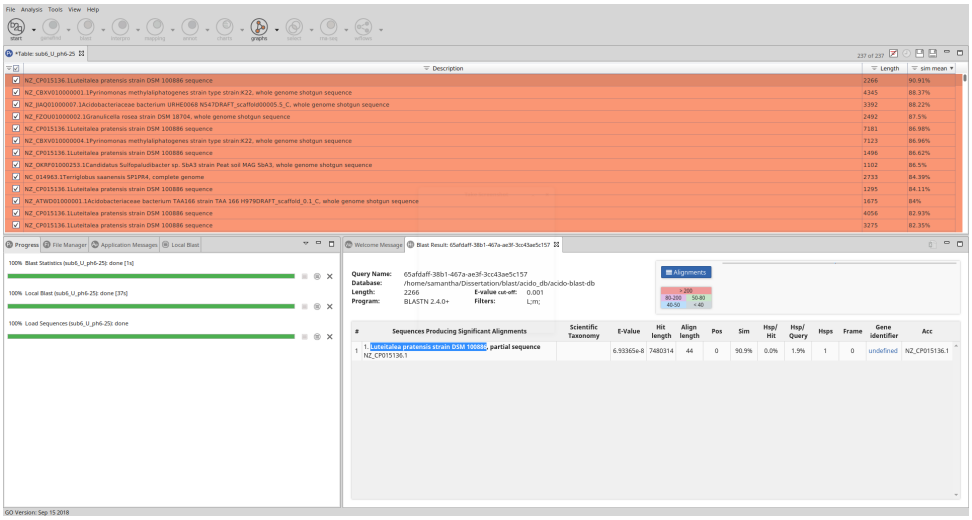
(a) *line* **(b)** *span*

Figure 8.2: `acidoseq` mean plots of the subdivisions based on the pH = 6.25, displaying subdivisions 4, 6, and 22 accordingly - plot type depends on user input into CLI.

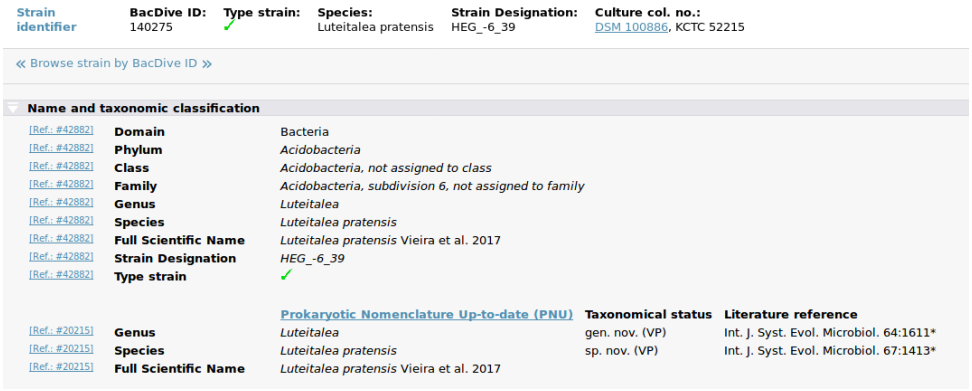
Figure 8.2 represents and visualises the outputs for the user: showing the subdivision `GC` mean for a collection of reads. Sub-figure 8.2a shows the average of mean scores; e.g. subdivision 4 genomes `GC` ratio ranges from ≈50% to ≈61% however all these averages result in an average of ≈60%. Sub-figure 8.2b specifically shows all the reads that will reside in which subdivision region. These plots differ depending on the pH: if a *low* pH of <5, subdivisions 1, 2, 3 and 13 will be plotted instead.

[1] Amplifying a specific DNA fragment to produce many sequence copies [95]

Subdivision 6 output file, `BLAST` and analysed



(a) `BLAST` against an Acidobacteria genome database using `Blast2Go`.



(b) Highest sequence similarity from the `BLAST` of subdivision 6 output.

Figure 8.3: Analysis of subdivision 6 output from `acidoseq`, gaining knowledge of what species subside in the subdivision based on the prediction.

Due to lack of `BLAST` resources, I decided to use `Blast2Go` on the `acidoseq` subdivision 6 output due to being the smallest file from a 6.25 pH. Highest sequence similarity was 90.91% (figure 8.3a): *Luteitalea pratensis*[2] (figure 8.3b), which resides in subdivision 6 of Acidobacteria: **proving my package works as expected**. Furthermore, *Pyrinomonas methylaliphatogenes strain type strain:K22*[3] was found, and this species resides in subdivision 4: as we could predict from the pH. An unclassified species was identified: *Acidobacteriaceae bacterium URHE0068*[4], this result can lead to further understanding and investigating that this unclassified genome could reside in subdivision 6, 4, or 23. From other studies, I found that after running a `BLAST` job, the output for certain subdivisions (the unclassified reads) seemed somewhat similar in their other pH placements; for example, a classified sequence being placed in subdivision 6, after `BLAST`, had some similarity for unclassified subdivision 4 reads - showing **evidence of the pattern similarity of Acidobacteria species within the pH levels**.

[2] `https://bacdive.dsmz.de/strain/140275`
[3] `https://www.ncbi.nlm.nih.gov/pubmed/26568784`
[4] `https://www.ncbi.nlm.nih.gov/nuccore/JIAQ00000000`

Part IV

Conclusion

Chapter 9

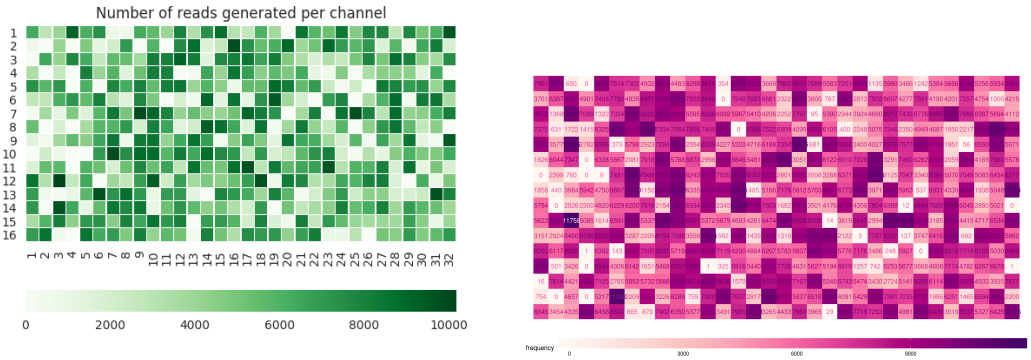
Discussion

This project has displayed that I have carried out a substantial amount of work: from a background study of Nanopore, a literature review of Acidobacteria, and studying the data-set from a variety of tools in addition to creating my own successfully - studying and using the bioinformatics tools proved useful for finding underlining issues: quality, for example quality decreases over time due to `MinIONQC` explaining that pores tire out but a ≈8hrs there's always a peak in quality due to changing pores.

From the different tools used from section 6.1, some limit us with a read-length/quality graph of maximum read-length of 10,000 (`pauvre` & `MinIONQC`) whilst others give us further details, yet limit the details; e.g. `NanoPlot` plots a maximum read-length of 50000< x <60000 on the axis but no actual plotting, whilst `POREquality` does plot 60000< x <65000 via another type of plot: histogram. We can conclude from this that the data does reach read-length heights of 60,000 yet tools limit the information: perhaps due to low quality. Each tool, though, does state the quality over 7, `MinIONQC` limits details to >=7, whilst `POREquality` and `pauvre` show a quality of 9 - furthermore `pauvre` displays most short-reads at 9 quality but it seems that long-reads are a score of 10 in quality. `NanoPlot` also displays an average quality of 9, but no other information provided for the longer reads: quality decreases over time of the MinION run but read-length production seems stable from their plot (sub-figure 6.2c).

Figure 9.1 shows a side-by-side comparison of `NanoPlot` and `POREquality`. We can observe there seems to be differences between the two: the plots are flipped vertically but can't be determine for sure; see sub-figure 9.1a (3,16) & (4,16) x/y axis and then flip vertically to observe the similarities of sub-figure 9.1b.

This study being an analysis of Acidobacteria and Nanopore data, yet tools designed for Nanopore data seem to differ yet also correlate: quality 9 seems to be consistent with the tools yet the plots all seem to display a different story. One would be best using all tools and choosing the most visually appealing, with substantial amount of information, to represent their data. Despite all this, we can't exactly compare these results with others due to lack of information on Nanopore sequencing, as stated in section 2.2. Plus there is a lack of Nanopore data-sets available.



(a) `NanoPlot` (b) `POREquality`

Figure 9.1: Comparison of reads generated per channel (512) of the Nanopore MinION through a heatmap/colour gradient scale.

Reflecting over the study, I have came to realise there is no package designed specifically for Acidobacteria; `acidoseq` is a great starter giving basic statistics on the coverage and read-length of the Acidobacteria classified reads. `acidoseq` requires a user to use `Kaiju` for fast classifications, which they can also look into what other species were identified in their sample without having to run a slow and long `BLAST` job.

Furthermore, there are no packages for Nanopore reads that look at `GC` specifically, while I understand many phylum don't depend on `GC` in their classes, it does seem to be a pattern in Acidobacteria. The `GC` plot `acidoseq` produces (figures 8.1 & 8.2) is a different style to `Goldilocks` (figure 6.7), `acidoseq` has more bins to get a better view of the count and the x-axis labels are less crowded. `Goldilocks`' scatter plot (sub-figure 6.7b) was designed for a single genome and short-reads and so perhaps the plot would be better with different data, however it did present the results: a high of ≈ 7.5 but overall the histograms from `acidoseq` essentially visualise this data but in a better layout.

I built up my Acidobacteria genome database from the NCBI assembly[1] section itself, whilst `NCBI_taxonomy_tree` method collects all data from NCBI with the `dmp` files plus extracting Acidobacteria genomes would prove to be a difficult task: possibly producing a collection of data which we don't want to use (memory wasting).

[1] via `https://www.ncbi.nlm.nih.gov/Taxonomy/TaxIdentifier/tax_identifier.cgi`

Chapter 10

Reflecting

Looking back over this project, I have learnt a lot about Acidobacteria and I believe I have achieved a lot. The biology of the data through the various techniques conducted shows that there is a substantial amount of Acidobacteria in the data-set despite the subsets by limiting via quality and read-length. To express this point again about Nanopore, Nanopore quality does actually seems to average around 10 - our reads however are heavily dominated by short, low quality. This may have affected some of our results throughout the project: lack of clusters/bins visualised.

I feel this dissertation displays a variety of skills learnt from my Masters degree: I used my knowledge of `Python` to create a package, `acidoseq`, in order to analyse data. My original technique was a sort of data-mining method in order to extract classified taxons that had specifically the term "Acidob" in order to gain "Acidobacterium" data (regular use of 'Acidobacterium' is used for unclassified reads), however I soon learnt and realised there were a wider variety of unclassified reads that had specific taxon names. But after improving my method, visualised the results, and output the data I can sufficiently say the package worked on our data: I was able to extract information from the sample and say that for a pH of 6.25 (Aberystwyth) subdivision 6 and 4 species, plus unclassified, were identified. If more Nanopore data were available, specifically metagenomes of soil samples, I would be able to explore soil data-sets in more detail.

From my background research, `acidoseq` is one of the first packages designed for Acidobacteria and one that looks into `GC` content. The user is given plenty of information regarding statistics and file output names. The plots are user-friendly allowing people to personalise the style. I originally attempted to combine the *line* and *span* plots with regions coloured and a bold line to show the average mean, however the results were cluttered and excessive. In future I think I would attempt another method in order to avoid users having to choose a plot style, or perhaps provide both plots to the user.

In future, once a new Countryside Survey is available I will update the pH across the package. pH has be seen to be rising steadily over periods of 10 years by $\approx$0.25 - $\approx$0.5 and so up-to-date figures will make the package more reliable. Moreover, with future additions of Acidobacteria genomes, plus partial sequences, I can study the `GC` content in more detail and so can develop `acidoseq` further with more precise knowledge of subdivision `GC` means.

The use of the `Kaiju` output file is great for science. I am promoting the use of different bioinformatics tools in order for researchers to gain the full potential and analysis of their data. Using `Kaiju` will visualise a user's data to observe the diversity. Potential developments, as one can read on `acidoseq` git `README.md` "Alter code so the input file can be the original Kaiju output" this future improvement would make `acidoseq` easier to use and avoids a user having to alter/edit the file first, which is currently a requirement. The task of making use of the original `Kaiju` output is one I did not originally indulge in due to not being a `csv` file (`kaiju.out`), but it could be a goal of choosing the correct columns from the file.

To conclude an overall view of this dissertation, it was successful. We found Acidobacteria in the sample from initial `Kaiju` job and found that we could assemble a longer, Acidobacteria sequence. We were also able to discover that Acidobacteria was present in the highest quality set of reads and obtained GO descriptions. Our research question was answered: I created a `Python` package that assigns unclassified Acidobacteria sequences into subdivisions based on the `GC` and pH. Furthermore, from the analysis of `acidoseq`'s results of the subdivision outputs we have proved that there is a pattern of `GC` and subdivision and proved the package works and we found subdivision 6 species from the subdivision 6 output file. We also found a pattern of similarity of Acidobacteria species within the same pH level. The `acidomap` feature proves useful as I could not find any current pH levels in Acidobacteria and so I chose the pH based on the visualisation, including other background research.

Part V

References

Bibliography

[1] K.-C. Wong, *Computational Biology and Bioinformatics: Gene Regulation.* CRC Press, 2016.

[2] J. M. Berg, J. L. Tymoczko, L. Stryer, *et al.*, "Biochemistry," 2002. pp. 118–19, 781–808.

[3] Alberts, B et al, "Molecular biology of the cel." Garland Science, 2014. 6th ed., Chapter 4: DNA, Chromosomes and Genomes.

[4] R. Shukla, *Analysis Of Chromosome.* Agrotech Press, 2014.

[5] J. A. Eisen, "Environmental shotgun sequencing: its potential and challenges for studying the hidden world of microbes," *PLoS biology*, vol. 5, no. 3, p. e82, 2007.

[6] T. Thomas, J. Gilbert, and F. Meyer, "Metagenomics-a guide from sampling to data analysis," *Microbial informatics and experimentation*, vol. 2, no. 1, p. 3, 2012.

[7] Cyfoeth Naturiol Cymru Natural Resouces Wales, "Metal mines strategy for wales," 2013. Available: `https://www.naturalresources.wales/media/1177/metal-mines-strategy-for-wales-september-2013.pdf?lang=en`, last accessed: 13/07/2018.

[8] "Model estimates of topsoil properties [countryside survey]," 2007. Owned by NERC (UK *National Environment Research Council* Centre for Ecology & Hydrology. Contains British Geological Survey material (copright 2014), and Ordinance Survey data (Crown copyright 2007).

[9] R. I. Griffiths, B. C. Thomson, P. James, T. Bell, M. Bailey, and A. S. Whiteley, "The bacterial biogeography of british soils," *Environmental microbiology*, vol. 13, no. 6, pp. 1642–1654, 2011.

[10] M. Bolland, C. Gazey, A. Miller, D. Gartner, and J.-A. Roche, "Subsurface acidity," *Department of Agriculture and Food, Western Australia Bulletin 4602*, 2004.

[11] M. K. Männistö, M. Tiirola, and M. M. Häggblom, "Bacterial communities in arctic fjelds of finnish lapland are stable but highly ph-dependent," *FEMS Microbiology Ecology*, vol. 59, no. 2, pp. 452–465, 2006.

[12] A. Eskelinen, S. Stark, and M. Männistö, "Links between plant community composition, soil organic matter quality and microbial communities in contrasting tundra habitats," *Oecologia*, vol. 161, no. 1, pp. 113–123, 2009.

[13] B. Emmett, B. Reynolds, P. Chamberlain, E. Rowe, D. Spurgeon, S. Brittain, Z. Frogbrook, S. Hughes, A. Lawlor, J. Poskitt, *et al.*, "Countryside survey: soils report from 2007," *NERC/Centre for Ecology and Hydrology*, 2010.

[14] I. Yolcubal, M. L. Brusseau, J. F. Artiola, P. J. Wierenga, and L. G. Wilson, "Environmental physical properties and processes," in *Environmental Monitoring and Characterization*, Elsevier Inc., 2004.

[15] BBC Radio 4, "Aberystwyth." `https://www.bbc.co.uk/programmes/b0b2gspb`, 2018. Online Podcast, 1h44 and 2h57, accessed: 2018-06-11.

[16] PSYCHROPATHS (Arwyn Edwards), "Zits & buried bodies – nanopore for radio." `https://psychropaths.wordpress.com/2018/05/17/zits-buried-bodies-nanopore-for-radio/`, 2018. Online retrieved from WordPress, accessed: 2018-06-11.

[17] A. Bhatia, J.-F. Maisonneuve, and D. H. Persing, "Propionobacterium acnes and chronic diseases," in *The Infectious Etiology of Chronic Diseases: Defining the Relationship, Enhancing the Research, and Mitigating the Effects: Workshop Summary., Knobler, SL et al.(eds.)*, pp. 74–80, 2004. Read via: `https://www.ncbi.nlm.nih.gov/books/NBK83685/`.

[18] H. Brüggemann, A. Henne, F. Hoster, H. Liesegang, A. Wiezer, A. Strittmatter, S. Hujer, P. Dürre, and G. Gottschalk, "The complete genome sequence of propionibacterium acnes, a commensal of human skin," *Science*, vol. 305, no. 5684, pp. 671–673, 2004.

[19] M. T. Madigan, J. M. Martinko, J. Parker, *et al.*, *Brock biology of microorganisms*, vol. 13. Pearson, 2017.

[20] T. J. Cameron and C. J. D., "Acidobacteria phyl. nov.," in *Bergey's Manual of Systematics of Archaea and Bacteria*, pp. 1–5, 2015. Read via: `https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118960608.pbm00001`.

[21] T. P. Niedringhaus, D. Milanova, M. B. Kerby, M. P. Snyder, and A. E. Barron, "Landscape of next-generation sequencing technologies," *Analytical chemistry*, vol. 83, no. 12, pp. 4327–4341, 2011.

[22] C. R. O'Donnell, H. Wang, and W. B. Dunbar, "Error analysis of idealized nanopore sequencing," *Electrophoresis*, vol. 34, no. 15, pp. 2137–2144, 2013.

[23] T. Laver, J. Harrison, P. O'neill, K. Moore, A. Farbos, K. Paszkiewicz, and D. J. Studholme, "Assessing the performance of the oxford nanopore technologies minion," *Biomolecular detection and quantification*, vol. 3, pp. 1–8, 2015.

[24] S. C. Pendleton, "An analysis of current state of the art software on nanopore metagenomic data," *bioRxiv*, 2018. 288969.

[25] S. M. Nicholls, W. Aubrey, A. Edwards, K. de Grave, S. Huws, L. Schietgat, A. Soares, C. J. Creevey, and A. Clare, "Computational haplotype recovery and long-read validation identifies novel isoforms of industrially relevant enzymes from natural microbial communities," *bioRxiv*, 2018. 223404.

[26] R. Lanfear, M. Schalamun, D. Kainer, W. Wang, and B. Schwessinger, "Minionqc: fast and simple quality control for minion sequencing data," *Bioinformatics*, p. bty654, 2018.

[27] S. J. Salter, M. J. Cox, E. M. Turek, S. T. Calus, W. O. Cookson, M. F. Moffatt, P. Turner, J. Parkhill, N. J. Loman, and A. W. Walker, "Reagent and laboratory contamination can critically impact sequence-based microbiome analyses," *BMC biology*, vol. 12, no. 1, p. 87, 2014.

[28] P. J. Cock, C. J. Fields, N. Goto, M. L. Heuer, and P. M. Rice, "The sanger fastq file format for sequences with quality scores, and the solexa/illumina fastq variants," *Nucleic acids research*, vol. 38, no. 6, pp. 1767–1771, 2009.

[29] D. J. Lipman and W. R. Pearson, "Rapid and sensitive protein similarity searches," *Science*, vol. 227, no. 4693, pp. 1435–1441, 1985.

[30] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin, "The sequence alignment/map format and samtools," *Bioinformatics*, vol. 25, no. 16, pp. 2078–2079, 2009.

[31] J. Gans, M. Wolinsky, and J. Dunbar, "Computational improvements reveal great bacterial diversity and high metal toxicity in soil," *Science*, vol. 309, no. 5739, pp. 1387–1390, 2005.

[32] L. Brussaard, "Biodiversity and ecosystem functioning in soil," *Ambio*, pp. 563–570, 1997.

[33] F. J. Meysman, J. J. Middelburg, and C. H. Heip, "Bioturbation: a fresh look at darwin's last idea," *Trends in Ecology & Evolution*, vol. 21, no. 12, pp. 688–695, 2006.

[34] M. Bundt, F. Widmer, M. Pesaro, J. Zeyer, and P. Blaser, "Preferential flow paths: biological 'hot spots' in soils," *Soil Biology and Biochemistry*, vol. 33, no. 6, pp. 729–738, 2001.

[35] A. M. Kielak, C. C. Barreto, G. A. Kowalchuk, J. A. van Veen, and E. E. Kuramae, "The ecology of acidobacteria: moving beyond genes and genomes," *Frontiers in microbiology*, vol. 7, p. 744, 2016.

[36] Thrash, J.C. and Coates, J.D., *Phylum XVII. Acidobacteria phyl.*, vol. 4. Bergey's Manual of Systematic Bacteriology, 2011. second edition. Springer, New York. Page 725.

[37] N. Kishimoto, Y. Kosako, and T. Tano, "Acidobacterium capsulatum gen. nov., sp. nov.: An acidophilic chemoorganotrophic bacterium containing menaquinone from acidic mineral environment," *Current Microbiology*, vol. 22, no. 1, pp. 1–7, 1991.

[38] S. M. Barns, E. C. Cain, L. Sommerville, and C. R. Kuske, "Acidobacteria phylum sequences in uranium-contaminated subsurface sediments greatly expand the known diversity within the phylum," *Applied and environmental microbiology*, vol. 73, no. 9, pp. 3113–3116, 2007.

[39] S. A. Eichorst, J. A. Breznak, and T. M. Schmidt, "Isolation and characterization of soil bacteria that define terriglobus gen. nov., in the phylum acidobacteria," *Applied and environmental microbiology*, vol. 73, no. 8, pp. 2708–2717, 2007.

[40] P. H. Janssen, "Identifying the dominant soil bacterial taxa in libraries of 16s rrna and 16s rrna genes," *Applied and environmental microbiology*, vol. 72, no. 3, pp. 1719–1728, 2006.

[41] M. Sait, P. Hugenholtz, and P. H. Janssen, "Cultivation of globally distributed soil bacteria from phylogenetic lineages previously only detected in cultivation-independent surveys," *Environmental microbiology*, vol. 4, no. 11, pp. 654–666, 2002.

[42] A. Quaiser, T. Ochsenreiter, C. Lanz, S. C. Schuster, A. H. Treusch, J. Eck, and C. Schleper, "Acidobacteria form a coherent but highly diverse group within the bacterial domain: evidence from environmental genomics," *Molecular microbiology*, vol. 50, no. 2, pp. 563–575, 2003.

[43] W. Liesack, F. Bak, J.-U. Kreft, and E. Stackebrandt, "Holophaga foetida gen. nov., sp. nov., a new, homoacetogenic bacterium degrading methoxylated aromatic compounds," *Archives of Microbiology*, vol. 162, no. 1-2, pp. 85–90, 1994.

[44] D. B. Meisinger, J. Zimmermann, W. Ludwig, K.-H. Schleifer, G. Wanner, M. Schmid, P. C. Bennett, A. S. Engel, and N. M. Lee, "In situ detection of novel acidobacteria in microbial mats from a chemolithoautotrophically based cave ecosystem (lower kane cave, wy, usa)," *Environmental microbiology*, vol. 9, no. 6, pp. 1523–1534, 2007.

[45] R. Morita, *Bacteria in oligotrophic environments: starvation-survival lifestyle*, vol. 1. Chapman & Hall New York, 1997.

[46] E. Smit, P. Leeflang, S. Gommans, J. van den Broek, S. van Mil, and K. Wernars, "Diversity and seasonal fluctuations of the dominant members of the bacterial soil community in a wheat field as determined by cultivation and molecular methods," *Applied and Environmental Microbiology*, vol. 67, no. 5, pp. 2284–2291, 2001.

[47] N. Fierer, M. A. Bradford, and R. B. Jackson, "Toward an ecological classification of soil bacteria," *Ecology*, vol. 88, no. 6, pp. 1354–1364, 2007.

[48] N. L. Ward, J. F. Challacombe, P. H. Janssen, B. Henrissat, P. M. Coutinho, M. Wu, G. Xie, D. H. Haft, M. Sait, J. Badger, *et al.*, "Three genomes from the phylum acidobacteria provide insight into the lifestyles of these microorganisms in soils," *Applied and environmental microbiology*, vol. 75, no. 7, pp. 2046–2056, 2009.

[49] L. R. Cleveland, "Symbiosis between termites and their intestinal protozoa," *Proceedings of the National Academy of Sciences*, vol. 9, no. 12, pp. 424–428, 1923.

[50] R. E. White, *Principles and practice of soil science: the soil as a natural resource*. John Wiley & Sons - Blackwell Publishing, 2005. 4th Edition.

[51] R. T. Jones, M. S. Robeson, C. L. Lauber, M. Hamady, R. Knight, and N. Fierer, "A comprehensive survey of soil acidobacterial diversity using pyrosequencing and clone library analyses," *The ISME journal*, vol. 3, no. 4, p. 442, 2009.

[52] P. K. Wüst, B. U. Foesel, A. Geppert, K. J. Huber, M. Luckner, G. Wanner, and J. Overmann, "Brevitalea aridisoli, b. deliciosa and arenimicrobium luteum, three novel species of acidobacteria subdivision 4 (class blastocatellia) isolated from savanna soil and description of the novel family pyrinomonadaceae," *International journal of systematic and evolutionary microbiology*, vol. 66, no. 9, pp. 3355–3366, 2016.

[53] D. E. Koeck, A. Pechtl, V. V. Zverlov, and W. H. Schwarz, "Genomics of cellulolytic bacteria," *Current opinion in biotechnology*, vol. 29, pp. 171–183, 2014.

[54] J. C. Thrash and J. D. Coates, "Phylum xvii. acidobacteria phyl. nov.," in *Bergey's Manual® of Systematic Bacteriology*, pp. 725–735, Springer, 2010.

[55] S. Hogg, *Essential microbiology*. John Wiley & Sons, 2013.

[56] K. Todar, *Nutrition and growth of bacteria*. Todar's Online Textbook of Bacteriology, 2013.

[57] Hentges, DJ, *Anaerobes: General Characteristics*, vol. 4. Medical Microbiology, 1996. Chapter 17. University of Texas Medical Branch at Galveston. Available from: `https://www.ncbi.nlm.nih.gov/books/NBK7638/`.

[58] Madigan MT; Martino JM, *Brock Biology of Microorganisms*, vol. 11. Pearson, 2006. Page 136. ISBN 0-13-196893-9.

[59] H. Izumi, T. Nunoura, M. Miyazaki, S. Mino, T. Toki, K. Takai, Y. Sako, T. Sawabe, and S. Nakagawa, "Thermotomaculum hydrothermale gen. nov., sp. nov., a novel heterotrophic thermophile within the phylum acidobacteria from a deep-sea hydrothermal vent chimney in the southern okinawa trough," *Extremophiles*, vol. 16, no. 2, pp. 245–253, 2012.

[60] D. B. Johnson, "Biodiversity and ecology of acidophilic microorganisms," *FEMS microbiology ecology*, vol. 27, no. 4, pp. 307–317, 1998.

[61] C. R. Kuske, S. M. Barns, and J. D. Busch, "Diverse uncultivated bacterial groups from soils of the arid southwestern united states that are present in many geographic regions.," *Applied and Environmental Microbiology*, vol. 63, no. 9, pp. 3614–3621, 1997.

[62] J. M. Willey, L. Sherwood, C. J. Woolverton, *et al.*, *Prescott, Harley, and Klein's microbiology*. McGraw-Hill Higher Education New York, 2008.

[63] S. A. Eichorst, D. Trojan, S. Roux, C. Herbold, T. Rattei, and D. Woebken, "Genomic insights into the acidobacteria reveal strategies for their success in terrestrial environments," *Environmental microbiology*, vol. 20, no. 3, pp. 1041–1063, 2018.

[64] M. Sait, K. E. Davis, and P. H. Janssen, "Effect of ph on isolation and distribution of members of subdivision 1 of the phylum acidobacteria occurring in soil," *Applied and Environmental Microbiology*, vol. 72, no. 3, pp. 1852–1857, 2006.

[65] C. L. Lauber, M. S. Strickland, M. A. Bradford, and N. Fierer, "The influence of soil properties on the structure of bacterial and fungal communities across land-use types," *Soil Biology and Biochemistry*, vol. 40, no. 9, pp. 2407–2415, 2008.

[66] B. K. Jung, K. Sang-Dal, A. R. Khan, L. Jong-Hui, Y.-H. Kim, J. H. Song, H. Sung-Jun, and S. Jae-Ho, "Rhizobacterial communities and red pepper (capsicum annum) yield under different cropping systems," *International Journal of Agriculture and Biology*, vol. 17, no. 4, 2015.

[67] A. K. Bartram, X. Jiang, M. D. Lynch, A. P. Masella, G. W. Nicol, J. Dushoff, and J. D. Neufeld, "Exploring links between ph and bacterial community composition in soils from the craibstone experimental farm," *FEMS microbiology ecology*, vol. 87, no. 2, pp. 403–415, 2014.

[68] C. L. Lauber, M. Hamady, R. Knight, and N. Fierer, "Pyrosequencing-based assessment of soil ph as a predictor of soil bacterial community structure at the continental scale," *Applied and environmental microbiology*, vol. 75, no. 15, pp. 5111–5120, 2009.

[69] A. Naether, B. U. Foesel, V. Naegele, P. K. Wüst, J. Weinert, M. Bonkowski, F. Alt, Y. Oelmann, A. Polle, G. Lohaus, *et al.*, "Environmental factors affect acidobacterial communities below the subgroup level in grassland and forest soils," *Applied and environmental microbiology*, pp. AEM–01325, 2012.

[70] D. Mount, "Bioinformatics: Sequence and genome analysis," *Cold Spring Harbor Laboratory Press: Cold Spring Harbor, NY.*, vol. 2, 2004. ISBN 0-87969-608-7.

[71] G. O. Consortium, "The gene ontology project in 2008," *Nucleic acids research*, vol. 36, no. suppl_1, pp. D440–D444, 2007.

[72] J. A. Martin and Z. Wang, "Next-generation transcriptome assembly," *Nature Reviews Genetics*, vol. 12, no. 10, p. 671, 2011.

[73] G. Rossum, "Python tutorial, technical report cs-r9526," 1995. Centrum voor Wiskunde en Informatica (CWI), Amsterdam. Python Software Foundation, Python Language Reference. Available at `http://www.python.org`.

[74] R Core Team, *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014. Available at `http://www.R-project.org/`.

[75] W. De Coster, S. D'Hert, D. T. Schultz, M. Cruts, and C. Van Broeckhoven, "Nanopack: visualizing and processing long read sequencing data," *bioRxiv*, p. 237180, 2018.

[76] N. J. Loman and A. R. Quinlan, "Poretools: a toolkit for analyzing nanopore sequence data," *Bioinformatics*, vol. 30, no. 23, pp. 3399–3401, 2014.

[77] S. M. Nicholls, A. Clare, and J. C. Randall, "Goldilocks: a tool for identifying genomic regions that are 'just right'," *Bioinformatics*, vol. 32, no. 13, pp. 2047–2049, 2016.

[78] D. E. Wood and S. L. Salzberg, "Kraken: ultrafast metagenomic sequence classification using exact alignments," *Genome biology*, vol. 15, no. 3, p. R46, 2014.

[79] P. Menzel, K. L. Ng, and A. Krogh, "Fast and sensitive taxonomic classification for metagenomics with kaiju," *Nature communications*, vol. 7, p. 11257, 2016.

[80] B. D. Ondov, N. H. Bergman, and A. M. Phillippy, "Interactive metagenomic visualization in a web browser," *BMC bioinformatics*, vol. 12, no. 1, p. 385, 2011.

[81] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of molecular biology*, vol. 215, no. 3, pp. 403–410, 1990.

[82] A. Conesa, S. Götz, J. M. García-Gómez, J. Terol, M. Talón, and M. Robles, "Blast2go: a universal tool for annotation, visualization and analysis in functional genomics research," *Bioinformatics*, vol. 21, no. 18, pp. 3674–3676, 2005.

[83] B. Buchfink, C. Xie, and D. H. Huson, "Fast and sensitive protein alignment using diamond," *Nature methods*, vol. 12, no. 1, p. 59, 2014.

[84] S. Koren, B. P. Walenz, K. Berlin, J. R. Miller, N. H. Bergman, and A. M. Phillippy, "Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation," *Genome research*, pp. gr–215087, 2017.

[85] H. Li, "Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences," *Bioinformatics*, vol. 32, no. 14, pp. 2103–2110, 2016.

[86] R. R. Wick, M. B. Schultz, J. Zobel, and K. E. Holt, "Bandage: interactive visualization of de novo genome assemblies," *Bioinformatics*, vol. 31, no. 20, pp. 3350–3352, 2015.

[87] S. Lerat, A.-M. SIMAO-BEAUNOIR, and C. Beaulieu, "Genetic and physiological determinants of streptomyces scabies pathogenicity," *Molecular plant pathology*, vol. 10, no. 5, pp. 579–585, 2009.

[88] C. Ash, F. G. Priest, and M. D. Collins, "Molecular identification of rrna group 3 bacilli (ash, farrow, wallbanks and collins) using a pcr probe test," *Antonie van Leeuwenhoek*, vol. 64, no. 3-4, pp. 253–260, 1993.

[89] C. C. Laczny, C. Kiefer, V. Galata, T. Fehlmann, C. Backes, and A. Keller, "Busybee web: metagenomic data analysis by bootstrapped supervised binning and annotation," *Nucleic acids research*, vol. 45, no. W1, pp. W171–W179, 2017.

[90] W. De Coster, S. D'Hert, D. T. Schultz, M. Cruts, C. Van Broeckhoven, and B. Berger, "Nanopack: visualizing and processing long-read sequencing data," *Bioinformatics*, vol. 10, 2017.

[91] P. Henrys, A. Keith, D. Robinson, and B. Emmett, "Model estimates of topsoil ph and bulk density [countryside survey]," *NERC Environmental Information Data Centre*, 2012. Available: `https://doi.org/10.5285/5dd624a9-55c9-4cc0-b366-d335991073c7`.

[92] Li et al., "Pysam." `http://www.ncbi.nlm.nih.gov/pubmed/19505943`, 2009.

[93] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing In Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[94] P. Yakovchuk, E. Protozanova, and M. D. Frank-Kamenetskii, "Base-stacking and base-pairing contributions into thermal stability of the dna double helix," *Nucleic acids research*, vol. 34, no. 2, pp. 564–574, 2006.

[95] R. K. Saiki, D. H. Gelfand, S. Stoffel, S. J. Scharf, R. Higuchi, G. T. Horn, K. B. Mullis, and H. A. Erlich, "Primer-directed enzymatic amplification of dna with a thermostable dna polymerase," *Science*, vol. 239, no. 4839, pp. 487–491, 1988.

[96] J. D. Wuitschick and K. M. Karrer, "Analysis of genomic g+ c content, codon usage, initiator codon context and translation termination sites in tetrahymena thermophila," *Journal of Eukaryotic Microbiology*, vol. 46, no. 3, pp. 239–247, 1999.

Part VI

Appendix

Appendix A

Information

Tool Versions

Report created in LaTeXwith ShareLaTeX & Overleaf 2018

Software versions:

- SAMtools v1.9
- Python v2.7
- R v3.2.3
- pauvre v0.1.8
- NanoPlot v1.14.2
- POREquality v0.8
- MinIONQC v1.3.5
- Goldilocks v0.1.1
- Kraken v2.0.6
- Kaiju v1.6.2
- Krona v2.7
- BLAST v2.6.0+
- Blast2Go v5.2.0
- Diamond v0.9.22
- Canu v1.7
- Minimap2 v2.12
- Miniasm v0.3
- Bandage v0.8.1
- BusyBee v2.7
- NanoFilt v2.2.0
- Python v3.5
- acidoseq v1.3.6

Appendix B

Command Queries

pauvre

```
$ pauvre marginplot --fastq all.fq.gz

$ pauvre stats --fastq all.fq > pauvre_stats.txt
```

Listing 14: Command query for `pauvre`, creating the qualit `marginplot` and the `stats` table of quality/read length statistics.

NanoPlot

```
$ NanoPlot --fastq_rich all.fq.gz --plots kde
```

Listing 15: Command line query for `NanoPlot`. Using `FASTQ_rich` for more plots produced and using the `kde` design for all `FASTQ` files stored in `gz` format.

POREquality

```
$ Rscript -e "rmarkdown::render('~/git/POREquality/POREquality.Rmd', output_file=
paste('/media/samantha/Seagate\ Backup\ Plus\ Drive/nanopore/vcr/fastq/reports/ru
n.html',sep=''))" -i /media/samantha/Seagate\ Backup\ Plus\ Drive/nanopore/vcr/fa
stq/sequencing_summary.txt -o /media/samantha/Seagate\ Backup\ Plus\ Drive/nanopo
re/vcr/fastq/reports
```

Listing 16: Query for `POREquality` the `RMarkdown` script.

MinIONQC

```
$ Rscript MinIONQC.R -i sequencing_summary.txt -o output -s TRUE -p 2
```

Listing 17: `Rscript` Query for `MinIONQC` - the output is a directory.

Goldilocks

```
from goldilocks import Goldilocks
from goldilocks.strategies import GCRatioStrategy # GC
from goldilocks.strategies import NucleotideCounterStrategy # ACGT

sequence_data = {"dataset":
    {"file": "acido_reads_2018-07-28_22-28-17.fa.fai"}
}

# ACGT
g = Goldilocks(
    NucleotideCounterStrategy(["A", "C", "G", "T"]),
    sequence_data,
    length="1K",
    stride="1K",
    is_faidx=True,
    processes=4
)
g.plot(prop=True, tracks=["A", "C", "G", "T"]) # Line graph

# GC ratio
g = Goldilocks(GCRatioStrategy(), sequence_data, length="1K", stride="1K", is_faidx=True)
g.plot(title="GC Content") # Scatter
g.plot(bins=50, bin_max=1.0, prop=True, title="GC Content Profile") # Hist
```

Listing 18: `Goldilocks` script I produced for plotting the `NucleotideCounterStrategy` `ACGT` line graphs and `GCRatioStrategy` `GC` scatter and histogram.

Kaiju

```
$ grep -A1 'e098e435-b876-4f22-8d5b-87e94dbf9068' *.fastq
```

Listing 19: Extracting an example sequence from a set of `FASTQ` files. Sequence ID was labelled as unclassified from the `Kaiju` output file.

```
fastq_runid_1149bcfab0a6db595a4ab43f8f059f76e4b96aaf_0.fastq-TAAAATCAGTAGCCAGCGTT
CCGTTACGTATTGCTCCACCGTGGCCGAGCGGTCGGTGGCAGCAGCAGCGGCAGCGGAAACCGCGCGGCGAACCTCCC...
```

Listing 20: The sequence extracted from the query, see listing 19.

Above is an unclassified read; `BLAST`: *Streptomyces glaucescens strain GLA.O, complete genome* - Score 54.7 bits, Query cover 2%. Results presented where `Kaiju` noted them as unclassified, due to low hits.

Diamond

```
$ diamond blastx --db nr.dmd.dmnd --query all.fa --out diamond_all-nr_blastx --ou
tfmt 5 -k 1 --sensitive -p 16
```

Listing 21: Query in order to run `Diamond`.

Minimap2

```
$ minimap2 -x ava-ont all.fq scrminimap2.sge
```

Listing 22: `Minimap2` query.

Miniasm

```
$ miniasm -f all.fq aminiasm.sge
```

Listing 23: `Miniasm` query.

NanoFilt

```
$ gunzip -c all.fq.gz | NanoFilt -q 9 -l 5000 | gzip > trimmed_q9l500.fq.gz

$ gunzip -c all.fq.gz | NanoFilt -q 10 -l 2500 | gzip > trimmed_q10l2500.fq.gz

$ gunzip -c all.fq.gz | NanoFilt -q 12 | gzip > trimmed_q12.fq.gz
```

Listing 24: Filtering the `FASTQ` file with `NanoFilt`.

acidoseq

```
$ sed -n '1~4s/^@/>/p;2~4p' all.fq > all.fa
```

Listing 25: Command to convert `FASTQ` to `FASTA`.

```
$ cut -f3,7 tax_report.txt > acidobacteria_taxid.csv
```

Listing 26: Command to extract column 3 and 7 from a taxonomy report from NCBI of a collection of acidobacteria species names and taxon ID.

```
$ pip3 install matplotlib
```

Listing 27: An example of installing the module, `matplotlib`, in order to run `acidoseq` through a `Linux terminal`.

```
$ acidoseq --taxdumptype U --kaijufile result_seqid_taxon.csv --fastapath all.fa
--style fast --plottype span --ph 6.25
```

Listing 28: Running `acidoseq` to observe only unclassified reads.

```
$ acidomap --city Aberystwyth
```

Listing 29: Running `acidomap`.

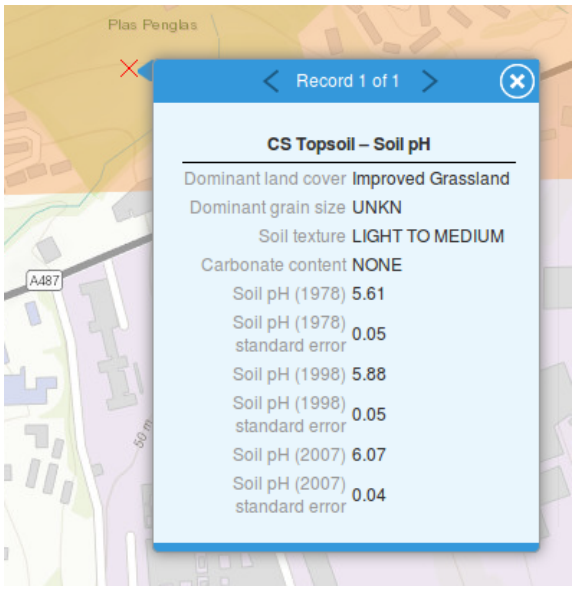


Figure B.1: Further evidence to back up using 6.25 as pH for my study: each 20 years the pH increases by 0.2 and a user can see from the map that Aberystwyth is in between 5-6.5 pH score.

Extracting from `Kaiju`

Steps to take in order to use the `Kaiju` output with my package, `acidoseq`.

```
$ awk '$1 == "C"' kaiju.out > kaijuC.out
```

Listing 30: Extracting all separate classified results from the `Kaiju` file.

```
$ cut -f2,3 kaijuC.out > results.txt
```

Listing 31: Cutting the second and third column of the `Kaiju` output file: the sequence IDs and corresponding NCBI taxonomy identifiers into a separate file.

```
$ sed 's/\s\+/,/g' results.txt > result_seqid_taxon.csv
```

Listing 32: Converted the `Kaju` output, classified filtered `txt` to `csv` (comma-delimted).

Converting `GFA` into `FASTA` for `acidoseq`

```
$ awk '/^S/{print ">"$2"\n"$3}' miniasm.gfa | fold > miniasm.fa
```

Listing 33: Query to convert `Miniasm GFA` to a `FASTA` for use with `acidoseq`.

Index