Taxonomic Assignment of 16S rRNA Sequences Based on Fourier Analysis

Guillermo G. Luque y Guzmán Sáenz[a], Alejandro Reyes Muñoz[a]

[a] *Research Group in Computational Biology and Microbial Ecology (BCEM) - Department of Biological Sciences, Universidad de los Andes*

Abstract

We introduce **TAXOFOR**, a novel machine learning classifier using *Random Forests* [Breiman, 2001] to assign taxonomy to paired-end sequencing amplicons up to genus level, trained with annotated sequences from the GreenGenes [DeSantis et al., 2006] database. It performs this task with a confidence close to 98% in terms of its accuracy, and it is faster than several of the *de facto* tools with the same purpose in microbial ecology. In order to manage the DNA sequences, at first they are numerically represented as projections into a 3D space defined by the vertex of a tetrahedron [Silverman and Linsker, 1986]. Afterwards, Discrete Fourier Transform allows to get their Power Spectra and use them as input both to train the classifier and to predict their taxonomy. Parseval's identity theorem ensures that similarity between the numerical representation of two DNA sequences can be gotten from their power spectra. This aspect is tested by comparing a dendrogram showing the results of a hierarchical clustering using the pairwise distance between the spectra of DNA sequences, with another one that

Email addresses: `gg.luque10@uniandes.edu.co` (Guillermo G. Luque y Guzmán Sáenz), `a.reyes@uniandes.edu.co` (Alejandro Reyes Muñoz)

has been built using the distance matrix obtained after a multiple sequence alignment (MSA). Performance and assertiveness of **TAXOFOR** against UCLUST [Ghodsi et al., 2011], RDP [Wang et al., 2007] and MOTHUR [Schloss et al., 2009] was assessed while assigning taxonomy to the same set of 16S rRNA sequences. The initial results are promising and give us enough room to implement improvements in terms of parallel processing and memory handling.

1. Introduction

Bacterial life is able to develop in diverse ecosystems, and given its abundance, it plays an essential role in multiple biochemical interactions [Pace, 1997], having a direct influence both in the surrounding environment as in the harboring host. Nay, composition of these communities in prokaryotic domain can be elucidated through the assessment of DNA sequences that code for the 16S ribosomal RNA subunit [Woese and Fox, 1977]. That is, seizing both the presence of highly conserved regions used as binding sites for specific primers; and the presence of hypervariable regions used to distinguish bacteria up to genus level [Chakravorty et al., 2007]. Overall, 16S rRNA analysis allows for the inference of phylogenetic structure between these organisms. In recent years, the use of next-generation sequencing (NGS) technologies has allowed us to avoid the problems with traditional culture-dependent microbial studies, where only a minimal percentage of microbes could be properly characterized, making possible the profiling of entire communities as in the realm of metagenomic studies [Grant and Long, 1981].

Among the different techniques to conduct metagenomic surveys in microbial ecology that take advantage of NGS deep sampling coverage, sequencing of 16S rRNA amplicons generated by PCR reactions can be considered one of the most used approaches [Sanschagrin and Yergeau, 2014], in spite of the biases introduced by specificity or coverage issues [Lilit Garibyan, 2014], or even primers selection [Soergel et al., 2012]. Once the data has been cleaned and preprocessed to reduce the number of sequencing errors, there are two different approaches to estimate the environmental sample's diversity.

The first of them, implies matching the sequences against a reference database for taxonomic assignment, but a correct assignation will depend on both the quality and the quantity of annotated sequences in the database [Lane et al., 1985a]. Yet, as the use of NGS technologies started to be ubiquitous in microbial ecology studies, there is a steadily increment in the number of new sequences databases, which at the same time makes the comparison of sequences an even more computationally expensive task.

In the second approach, sequences are clustered according to their similarity into operational taxonomic units (OTUs) [Liu et al., 2008]. Later, representative sequences will be chosen from each of these OTUs and compared to a reference database to perform their taxonomy assignment. One advantage of this approach is that it allows to identify novel sequences in contrast of the taxonomy dependent approach. However, several factors could influence the obtained results such as: the selection of a clustering method from between hierarchical, heuristic or model based [Lane et al., 1985b]; the similarity threshold, which is generally set to 97% nucleotide identity without having a complete biological sense; and the calculation of the distance matrix, that can be done using either multiple sequence alignments (MSA) or alignment-free methods [Schloss and Westcott, 2011].

In fact, variation in alignment quality can have a significant effect on the estimated diversity [Schloss, 2010], moreover when MSA calculated distances are amplified by the constraint of preserving homology across the set of sequences. Besides, most of the frequently alignment-free methods used for sequence comparison are based on a type of feature extraction that could lead to lose structural information or simply not taking it into account. One of these methods is related to counting the frequencies of k-length fixed words within the sequences we are analyzing, which has the additional trouble of being highly sensitive to the chosen value for k in terms of computational performance.

Taking this facts into account, we want to tackle the problem with a completely different approach. What if we consider that a non-coding DNA sequence may be interpreted as a discrete non-periodic signal? If so, from a signal processing perspective, this biological signal have to be composed by a finite number of observations (its nucleotides) in time or space domain. Ergo, we could use a Discrete Fourier Transform (DFT) to get the list of complex coefficients of a finite combination of complex sinusoids ordered by the frequencies present in the original signal.

Because these coefficients are complex numbers, it is preferable to get a Power Spectrum Density (PSD) which describes in a better way the distribution of frequency components composing the signal squaring their absolute values. Spectral analysis of DNA sequences are useful to detect any latent or periodical signal in them as for example approximate repeats of nucleotides. Indeed, a peak in the k-th position of a PSD signal indicates that a nucleotide tends to appear about N/k positions, being N the length of the sequence.

As PSD conveys the nucleotide distribution information of the original sequence, it is reasonable to think of it as a way to compare and classify DNA sequences. This idea is widely illustrated and very supported by several studies such us ([Yin and Wang, 2014], [Yin et al., 2014], [Yin and Yau, 2015] or [King et al., 2014]); that have shown the utility of Discrete Fourier Transform (DFT) to improve the classification of DNA sequences preserving all their related information, especially in cases where sequences undergo rearrangements during events involving homologous recombination [Teyssier et al., 2003]. In contrast to the mentioned studies, we use a different numerical mapping of DNA sequences and seize the resulting power spectra as discriminating features for our proposed classifier.

Here we present a software for taxonomic assignment of 16S rRNA sequences using Fourier analysis. This software is a machine learning classifier trained in a supervised fashion with labeled sequences from the GreenGenes [DeSantis et al., 2006] database and, it is based on an ensemble learning method called *Random Forests* [Breiman, 2001]. It can assign taxonomies to DNA sequence fragments enclosed by selected primer pairs that have been extensively used in the amplification of a broad range of phylotypes in varied community samples. Each sequence that it is presented to the classifier, is projected onto a orthogonal space where structural information might be preserved. In order to do that, the DNA sequence is considered as a composition of three binary signals using a vertex projection of each nucleotide. After obtaining a numerical version of a DNA fragment, it is appropriate to apply Fourier Analysis to it in order to get the Power Spectrum and use them for both training and assignation of taxonomy.

2. Background

2.1. Fourier Analysis

Fourier analysis allows for the study of a function or signal $f(t)$ that characterizes an observed phenomenon from its constituent parts, moving our understanding of it from a time or space domain onto a frequency domain. The function in question may exhibit a regularly repeating pattern either in time or space. It is worth to notice that $f(t)$ is periodic of period T if there is a number $T > 0$ such that $f(t + nT) = f(t)$, with $n \in \mathbb{N}$. Using a *Fourier series*, a periodic function $f(t)$ of period T, rewritten as $f(s)$ with $s = Tt$, can be expanded into a infinite summation of complex exponentials as in eq. (1), whenever $f(t)$ is in $L^2([0, 1])$ i.e. it is square-integrable in the interval $[0, 1] \in \mathbb{R}$ and has finite energy.

$$f(s) = \sum_{n=-\infty}^{\infty} \hat{f}(n) e^{2\pi ins/T} \tag{1}$$

The terms $\hat{f}(n)$ are called the *Fourier coefficients* of $f(t)$ and are given by eq. (2).

$$\hat{f}(n) = \int_{-T/2}^{T/2} e^{-2\pi ins/T} f(s)\, ds \tag{2}$$

Therefore, being able to write $f(t)$ as a Fourier series implies that it is synthesized from many positive and negative frequencies that conform its *spectrum.* If the period of $f(t)$ is T, then the frequencies in its spectrum are evenly spaced $1/T$ apart. This fact points to a reciprocal relation between the time domain and the frequency domain. In a complementary manner, the *power spectrum* defined by the set of squared magnitudes $|\hat{f}(n)|^2$ of the Fourier coefficients eases the graphical representation of the spectrum, giving a way of comparing two signals. In addition, the Fourier expansion won't always be an infinite sum, provided that $\hat{f}(n) = 0$ for any $n \in \mathbb{N}$ such as $|n| > N$; in this case it is said that $f(t)$ is bandlimited, having a bandwidth of N.

2.2. Fourier Transform

The *Fourier transform* is defined as an operation $\mathscr{F}$ applied to a nonperiodic signal $f(t)$ producing a complex valued function $\hat{f}(s)$ for any $s \in \mathbb{R}$, according to eq. (3).

$$\mathscr{F}\hat{f}(s) = \int_{-\infty}^{\infty} e^{-2\pi ist} f(t)\, dt \tag{3}$$

Conversely, through the inverse Fourier transform $\mathscr{F}^{-}$ given by eq. (4) we can recover the original signal $f(t)$ from its transform $\hat{f}(s)$

$$\mathscr{F}^{-1} f(t) = \int_{-\infty}^{\infty} e^{2\pi ist} \hat{f}(s)\, ds \tag{4}$$

Unlike Fourier series, the spectrum of a nonperiodic signal is a continuum of frequencies rather than a discrete set of integers. Even so, we have a power spectrum similarly defined to the periodic case. Besides, the Parseval's identity for Fourier transform states an important relation shown by eq. (5) between the energy of the function in the time domain and the power spectrum in the frequency domain.

$$\int_{-\infty}^{\infty} |f(t)|^2 \, dt = \int_{-\infty}^{\infty} |\hat{f}(s)|^2 \, ds \tag{5}$$

2.3. Discrete Fourier Transform

Beyond the periodicity issue, both Fourier series and Fourier transform are used to analyze functions of a continuous variable. But data and its measurements in the real world, such as in the case of a DNA sequence, are perceived in discrete form. Hence, the *discrete Fourier transform* (DFT) $\underline{\mathscr{F}}$ converts an N-tuple function $\mathbf{f} = (f[0], f[1], ..., f[N-1])$ representing a discrete input into an N-tuple $\mathbf{F} = (F[0], F[1], ..., F[N-1])$ output according to eq. (6).

$$\mathbf{F} = \underline{\mathscr{F}}\mathbf{f} = \sum_{n=0}^{N-1} \mathbf{f}[n]\mathbf{w}^{-n} \tag{6}$$

where $\mathbf{w} = (1, w^1, ..., w^{N-1})$, $w = e^{-2\pi i/N}$, and the m-Th output's component is given by eq. (7).

$$\mathbf{F}[m] = \sum_{n=0}^{N-1} \mathbf{f}[n] e^{-2\pi i m n/N},\ m = 0, 1, ..., N-1 \tag{7}$$

The inverse DFT $\underline{\mathscr{F}}^{-1}$ relies on the discrete orthogonality of the complex exponentials and is defined by eq. (8).

$$\mathbf{f} = \underline{\mathscr{F}}^{-1}\mathbf{F} = \frac{1}{N}\sum_{n=0}^{N-1} \mathbf{f}[n]\mathbf{w}^{n} \tag{8}$$

Due to the periodicity of the complex exponentials in eq. (6), both input and output of DFT must be considered as periodic functions of period N; thus DFT can be defined over any range of N consecutive indexes. As another interesting feature of the DFT, the spectrum of the input signal is splitted at its midpoint; so by convention, the first half of the spectrum is linked to the positive frequencies and the second half to the negative frequencies. Whichever discrete input **f**, all the information in its spectrum is in the first component $F[0]$ (the sum of the components in the input), the components $F[1], ..., F[N/2-1]$), and in the $F[N/2]$ component (the alternating sum of the components in the input).

3. Methods and Algorithms

3.1. Power spectrum of a DNA sequence

A DNA sequence S is a finite succession of N symbols $S_0 S_1 ... S_{N-1}$ from a fixed alphabet $\Sigma = \{A, C, G, T\}$ which reflects the order of nucleotides (Adenine, Cytosine, Guanine, and Thymine respectively) within a DNA molecule.

In order to make them computationally tratable, a number of numerical representation methods have been proposed which can be broadly organized depending on either the use of a fixed mapping or a physico-chemical property based mapping [Kwan and Arniker, 2009]. From the extensive set of mapping techniques, we decided to use the Voss representation [Voss, 1992], under which S can be rewritten as a linear combination of 4 binary indicator sequences $b_{i\in\Sigma}$ in such a way that,

$$x[n] = b_A[n] + b_C[n] + b_G[n] + b_T[n] \tag{9}$$

with $n = 0, 1, ..., N-1$, and $b_i[n]$ could take the value of either one or zero at position n depending on whether or not the element $S[n]$ has the same symbol than the pointed by i. Therefore, we can apply eq. (6) to calculate the power spectrum PS_S of a DNA sequence as follows:

$$PS_S[n] = |F_A[n]|^2 + |F_C[n]|^2 + |F_G[n]|^2 + |F_T[n]|^2 \tag{10}$$

where:

$$F_i[n] = \sum_{n=0}^{N-1} b_i[n]\mathbf{w}^{-n},\ i \in \Sigma \tag{11}$$

Another fixed mapping representation we evaluated was the proposed by [Silverman and Linsker, 1986], which formulates that S can also be mapped onto a 3D space as of associating each of the symbols in Σ to a vertex of a regular 3-simplex or tetrahedron. Each vertex consists on a vector in $\mathbb{R}^3$ with norm equal to 1, as for example:

$$\begin{aligned}
A &\rightarrow (a_r, a_g, a_b) = \mathbf{k} \\
C &\rightarrow (c_r, c_g, c_b) = \frac{-\sqrt{2}}{3}\mathbf{i} + \frac{\sqrt{6}}{3}\mathbf{j} - \frac{1}{3}\mathbf{k} \\
G &\rightarrow (g_r, g_g, g_b) = \frac{-\sqrt{2}}{3}\mathbf{i} - \frac{\sqrt{6}}{3}\mathbf{j} - \frac{1}{3}\mathbf{k} \\
T &\rightarrow (t_r, t_g, t_b) = \frac{2\sqrt{2}}{3}\mathbf{i} - \frac{1}{3}\mathbf{k}
\end{aligned}$$

Thus, the original DNA sequence is decomposed into three numerical sequences χ_r, χ_g, and χ_b of the same length N:

$$\begin{aligned}
\chi_r[n] &= \frac{\sqrt{2}}{3}(-b_C[n] - b_G[n] + 2b_T[n]) \\
\chi_g[n] &= \frac{\sqrt{6}}{3}(b_C[n] - b_G[n]) \\
\chi_b[n] &= \frac{1}{3}(3b_A[n] - b_C[n] - b_G[n] - b_T[n])
\end{aligned}$$

Once $\chi_{i\in\{r,g,b\}}[n]$ has been obtained, the power spectrum of this representation PS_S^{3D} is given by eq. (12).

$$PS_S^{3D}[n] = |F_r[n]|^2 + |F_g[n]|^2 + |F_b[n]|^2 \tag{12}$$

where:

$$F_i[n] = \sum_{n=0}^{N-1} \chi_i[n]\mathbf{w}^{-n},\ i \in \{r, g, b\} \tag{13}$$

Although it only indicates the frequencies of the nucleotides in a sequence, tetrahedron representation is recognized as a suitable representation for spectral analysis of DNA sequences [Anastassiou, 2001]. All the more, it has been shown that the power spectra obtained from the two tested techniques are essentially the same [Coward, 1997].

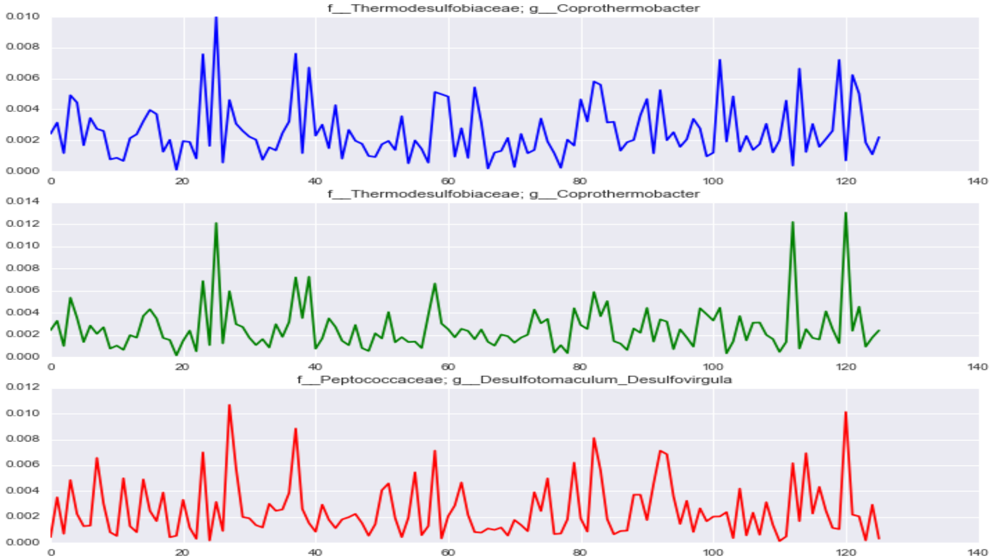

Figure 1: Power spectrum of the numerical representations of three different sequences from *Firmicutes* phylum.

For example, in fig. 1 is plotted the power spectra of three different sequences from *Firmicutes* phylum using the proposed numerical mapping. In particular, these sequences have a length of 253bp and correspond to fragments flanked by the pair of primers E517F and U806R. The first and second spectrum have in common energy peaks on several ranges of frequencies (e.g. 20hz-30hz, 35hz-40hz, 110hz-120hz), which in fact reflects the similarity between the original DNA sequences since both belong to the same family. In contrast, the third one has different number of distinctive peaks in the same intervals. However, after a closer inspection it can be observed a kind of equivalence in the amplitude of Fourier coefficients linked to the frequencies 32hz, 115hz and 120hz approximately.

3.2. Measuring similarity

Similarity between the numerical representation of two DNA sequences can be obtained from their power spectra. In fact, due to Parseval's identity (eq. (5)) and if it is deemed that DFT is a linear transformation, an L^2 distance of two signals into the time/space domain is equivalent to the same L^2 distance in their frequency domain, which it is given by eq. (14).

All the more, L^2 distance is appropriate to estimate similarity, as it is preserved under orthonormal transformations (like DFT) and is tolerant to additive Gaussian noise. Other metrics distinct to L^2 were considered, such as cosine, correlation or spectral information divergence but their computational cost was slightly superior to L^2 but had a similar discrimination value.

$$L^2(x,y) = \big(\sum_i (x_i - y_i)^2\big)^{\frac{1}{2}} \tag{14}$$

3.2.1. Akima interpolation

However, L^2 distance is restricted to points in the same N space, what means that the power spectra of the measured DNA numerical sequences have to be of equal length. This restriction impedes the direct application of L^2 distance, due to the fact that the length of the power spectrum depends only on the length of the transformed signal. Even though several approaches in the field of signal processing has been used to untangle this situation, such as linear interpolation [Yin and Wang, 2014], modulo-N reduction [Orfanidis, 2009] or using the last few Fourier coefficients [Rafiei and Mendelzon, 1998], we decided to use a more effective way to equate the lengths of the power spectra to make them comparable in an L^2 space.

The interpolation method stated by [Akima, 1970], commonly known as *Akima interpolation*, permits the fitting of the power spectra onto numeric vectors of fixed length, without introducing any distortion that could have led to a appreciable difference whether in its shape (fig. 2) or the amount of energy. From a coarse view, this method consists of successively applying a piecewise function using third order polynomials to each point of the data set. All of this is done in such a way that the slope of the curve defined by each polynomial is determined locally using a set of the nearest neighbor points to the point in question.

3.3. Dendrogram construction

Power spectra, eventually stretched, computed on the suggested numerical mapping of a set of DNA sequences might be used to build a pairwised distance matrix using an L^2 metric. Straight away, a neighbor-joining clustering method [Saitou N, 1987] allows for the construction of a dendrogram

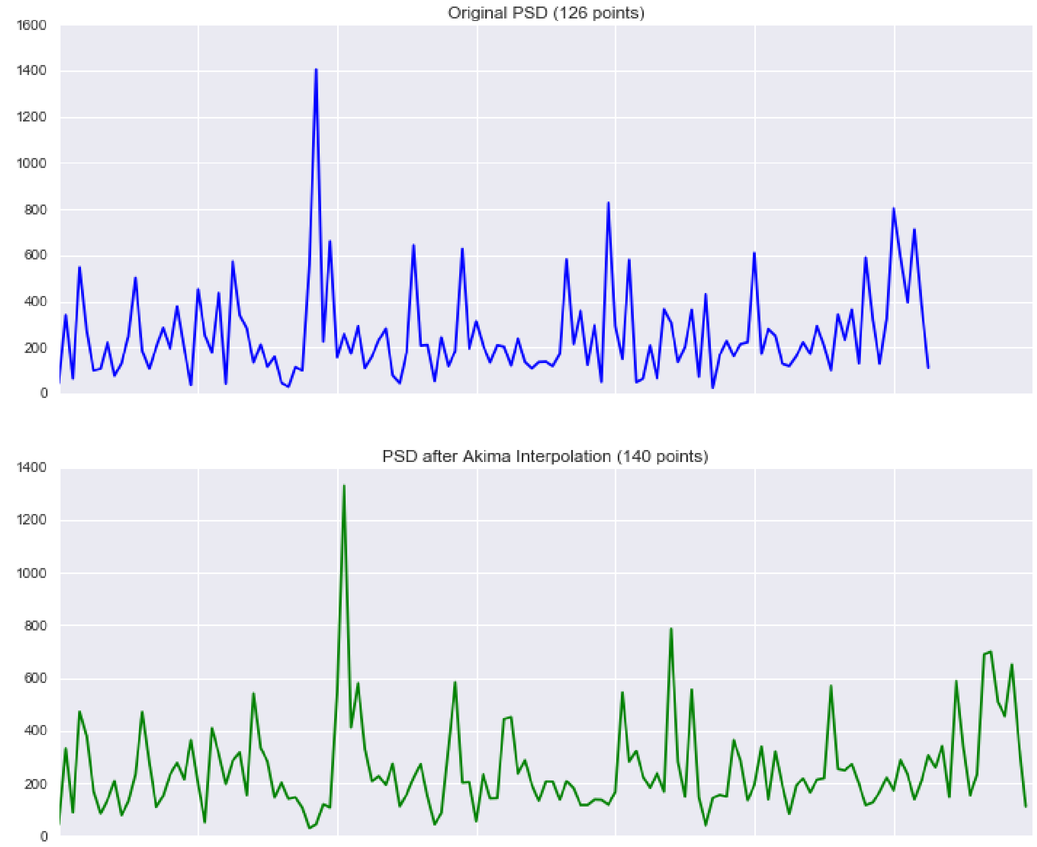

Figure 2: *Above*.- In blue, power spectrum of the region between E517F and U806R primers from a species of *Comamonadaceae Aquabacterium* genus, which has a length of 253bp. *Below*.- In green, the same spectrum after being stretched using the Akima interpolation.

seizing the generated matrix. Regarding the stretching process, the lengthening measure m is given for the length of the largest power spectrum in the set. We added up all of these points into the algorithm [1].

On the other hand, we performed a multiple sequence alignment with CLUSTAL [Chenna et al., 2003] of the same set of DNA regions, and used the alignment results to calculate a pairwise distance similarity matrix. Next, the dendrograms generated by our algorithm were compared to those obtained by applying a neighbor-joining clustering using the distance matrix with auspicious results as it is covered later.

3.4. Machine Learning Classification

We have implemented a machine learning predictive model to assign a 16S rRNA gene sequence. In particular, the fragment that corresponds to a DNA sequence flanked by a specific pair of forward and reverse primers. Our model takes advantage of *Random Forests* i.e. an ensemble of decision trees where each of them is built from a sample drawn with replacement, and using a random feature selection during its conformation [Breiman, 2001]. Induced randomness makes that bias in this type of models may have a slight increment in comparison to a normal decision tree, but the process of averaging the trained trees compensates this increase with a decrease in the variance and, in consequence, leading to better predictions.

3.4.1. Identifying regions of interest

One of our main concerns was to build a classifier as robust and accurate as to be useful in a bacterial 16S rRNA amplicon classification, so the model was trained in a supervised way with a subset of sequences from the GreenGenes Database that meet the following conditions: they were annotated up to genus level, and they presented what we called regions of interest i.e. regions limited by known primer pairs. There are 1.262.986 sequences in the version 13.5 of the GreenGenes Database, but only the 93.10% of them has non degenerated IUPAC characters, and are part of the *Bacteria* domain. Despite, in this bacterial set of sequences there are about 20.77% of redun-

Algorithm 1: Dendrogram construction.

```
input  : A set $S$ of $p$ DNA sequences
output: A dendrogram $T$
begin
    $PS \leftarrow$ an empty list
    for $i \leftarrow 1$ to $p$ do
        $\chi_{l \in \{r,g,b\}} \leftarrow$ tetrahedron_mapping($S[i]$)
        $F_i[n] \leftarrow \sum_{n=0}^{N-1} \chi_l[n]\mathbf{w}^{-n}$, $l \in \{r, g, b\}$
        $PS[i] \leftarrow \sum_{l \in \{r,g,b\}} |F_i[l]|^2$
        if $n$ is even then
            $t \leftarrow int(n/2)$
        else
            $t \leftarrow int(n/2) + 1$
        $PS[i] \leftarrow PS[i][1..t]$
    $m \leftarrow$ max length $PS$
    for $i \leftarrow 1$ to $p$ do
        $n \leftarrow$ length $PS[i]$
        if $n < m$ then
            $PS[i] \leftarrow$ akima_interpolation($PS[i], m$)
    for $i \leftarrow 1$ to $p$ do
        for $j \leftarrow 1$ to $p$ do
            if $i \neq j$ then
                $M[i, j] \leftarrow L^2(PS[i], PS[j])$
            else
                $M[i, j] \leftarrow 0$
    $T \leftarrow$ neighbor joining clustering using $M$
```

dant sequences at 100%, which in general dovetails with sequences stored in the database having distinct identification but the same assignment. In this research, our predictive model is centered on non-repeated bacterial sequences that were annotated up to genus level in GreenGenes, which number is reckoned at 613.493 sequences.

On the other hand, the set of known primer pairs was restricted to those used for paired-end 16s community sequencing on the Illumina HiSeq considered both in [Soergel et al., 2012] as well in [Caporaso et al., 2012]. We performed a search with regular expressions of a number of primer pairs on the GreenGenes' bacterial sequences without degenerated symbols, including those with repetitions, and with a defined phylum. In total, 22 forward primers and 23 reverse primers on 1.175.461 sequences distributed in 73 phyla were identified.

Name	Type	IUPAC Sequence
U515F	Fwd	GTGYCAGCMGCCGCGGTAA
E341F	Fwd	CCTACGGGNGGCNGCA
E517F	**Fwd**	**GCCAGCAGCCGCGGTAA**
U806R	**Rev**	**GGACTACNVGGGTWTCTAAT**
E926Ra	Rev	CCGNCNATTNNTTTNAGTTT
E1406R	Rev	GACGGGCGGTGWGTRCA
E533Ra	Rev	TNACCGNNNCTNCTGGCAC

Table 1: Primers chosen after inspecting bacterial sequences in GreenGenes Database. In bold, the primers used to build the classifier.

After measuring each primer's coverage per phylum, i.e. the percentage of sequences from a given phylum where a primer was found either in $5' - 3'$ direction or in their reverse complement, a subgroup of primers with the best results was used to generate the heat map in fig. 3. Later, the counting of the number of occurrences of each selected primer pair allows for the generation of the heat map shown in fig. 4, where it has been additionally marked the primer pair composed by E517F and U806R which were used to build the prototype classifier. The set of selected primers is described in table 1. They were chosen due to their coverage and because they present a well balanced sequence length among their enclosed regions. In fig. 5 can be seen how many sequences of a given length were matched by any of the selected primer pairs. As it was pointed earlier, we have focused on assigning taxonomy to sequences flanked by **E517F** and **U806R** in this research.

Therefore, regions enclosed by the primer pair E517F-U806R in bacterial sequences annotated up to genus level in GreenGenes database, were picked to be preprocessed and used in the training of our classifier. This owing to that pair of primers was found in 542.808 sequences of this kind without repetitions, and near 99% of the regions they flank had a length of roughly 253bp, and close to 280bp in less than 1%. Notwithstanding, although we are using the primer pair mentioned before for the extent of this research, our method is not restricted to it. Actually, we can train our classifier with whatever primer pair commonly used in paired end sequencing for instance, but its performance in term of accuracy is going to be affected by the number of occurrences of the primer pair along the reference database.

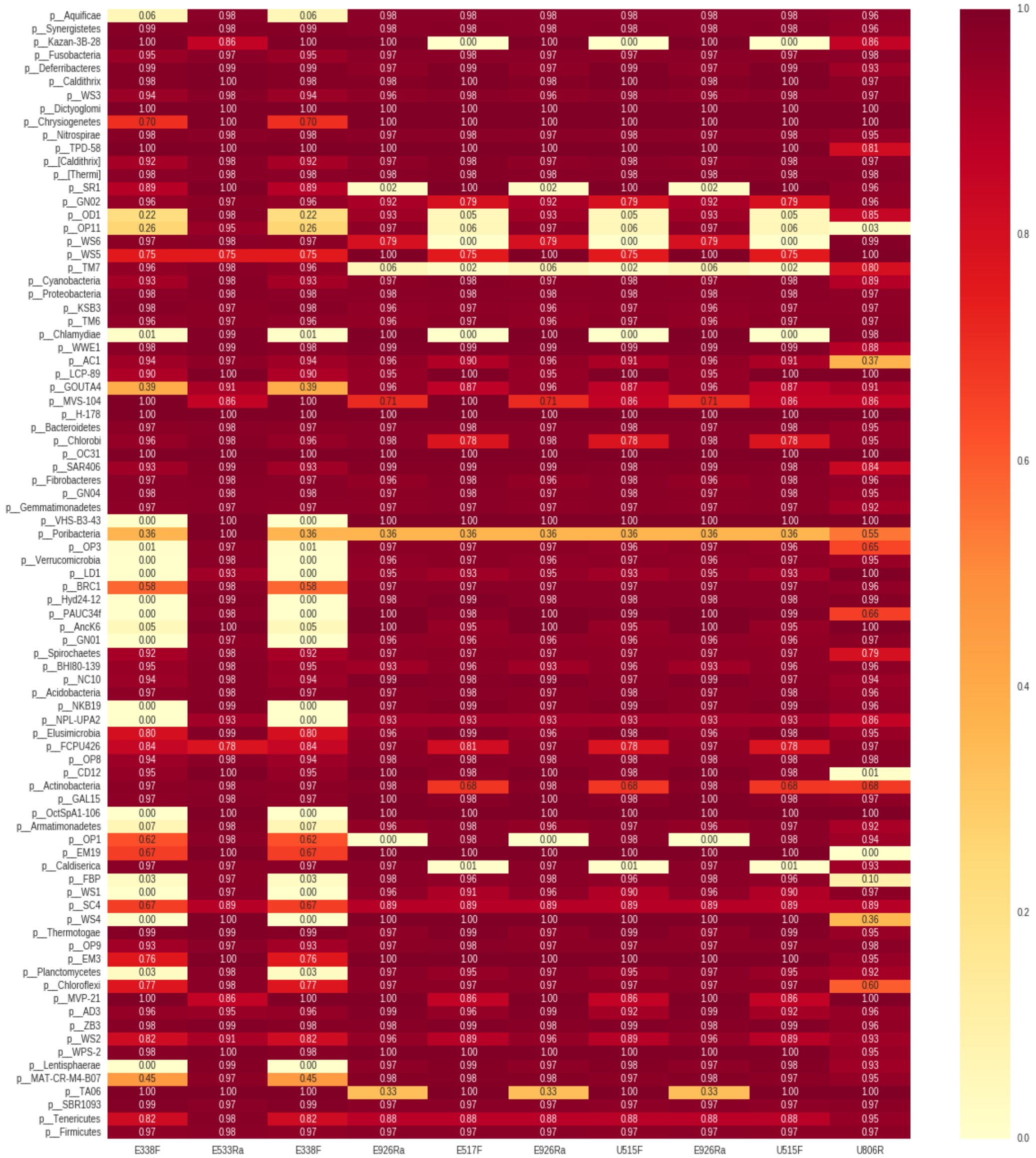

Figure 3: Heat map comparing the percentage of coverage among the studied phyla of a set of candidate primers (both forward and reverse). Each primer is located one after the other along (X-axis) and the phyla (Y-axis) are sorted by their evolutionary distance.

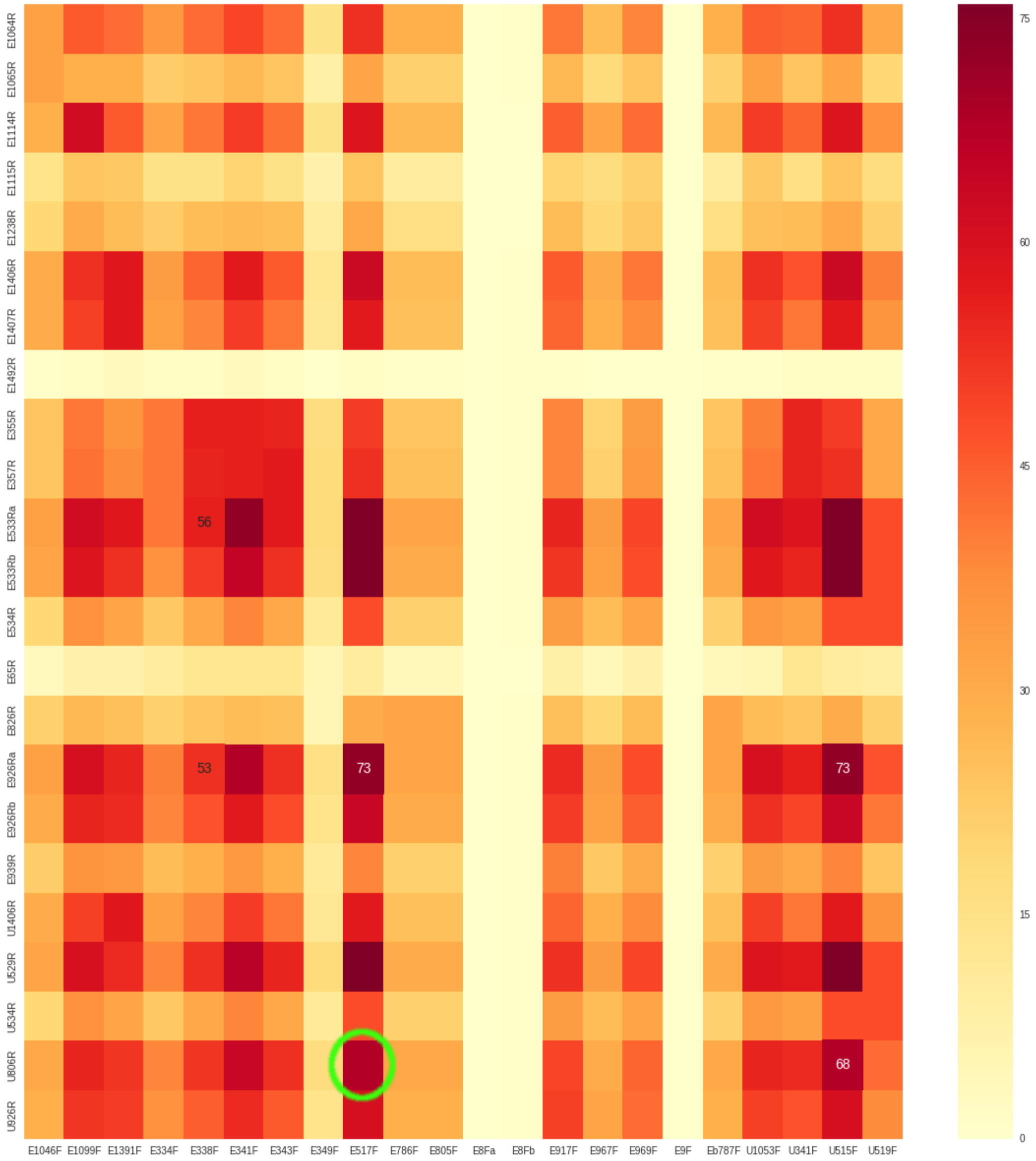

Figure 4: Heat map comparing the percentage of sequences from GreenGenes Database that contain a specific primer pair. Axis represent commonly used Forward (Y-axis) and Reverse (X-axis) 16S primers. A green circle surrounds E517F-U806R.

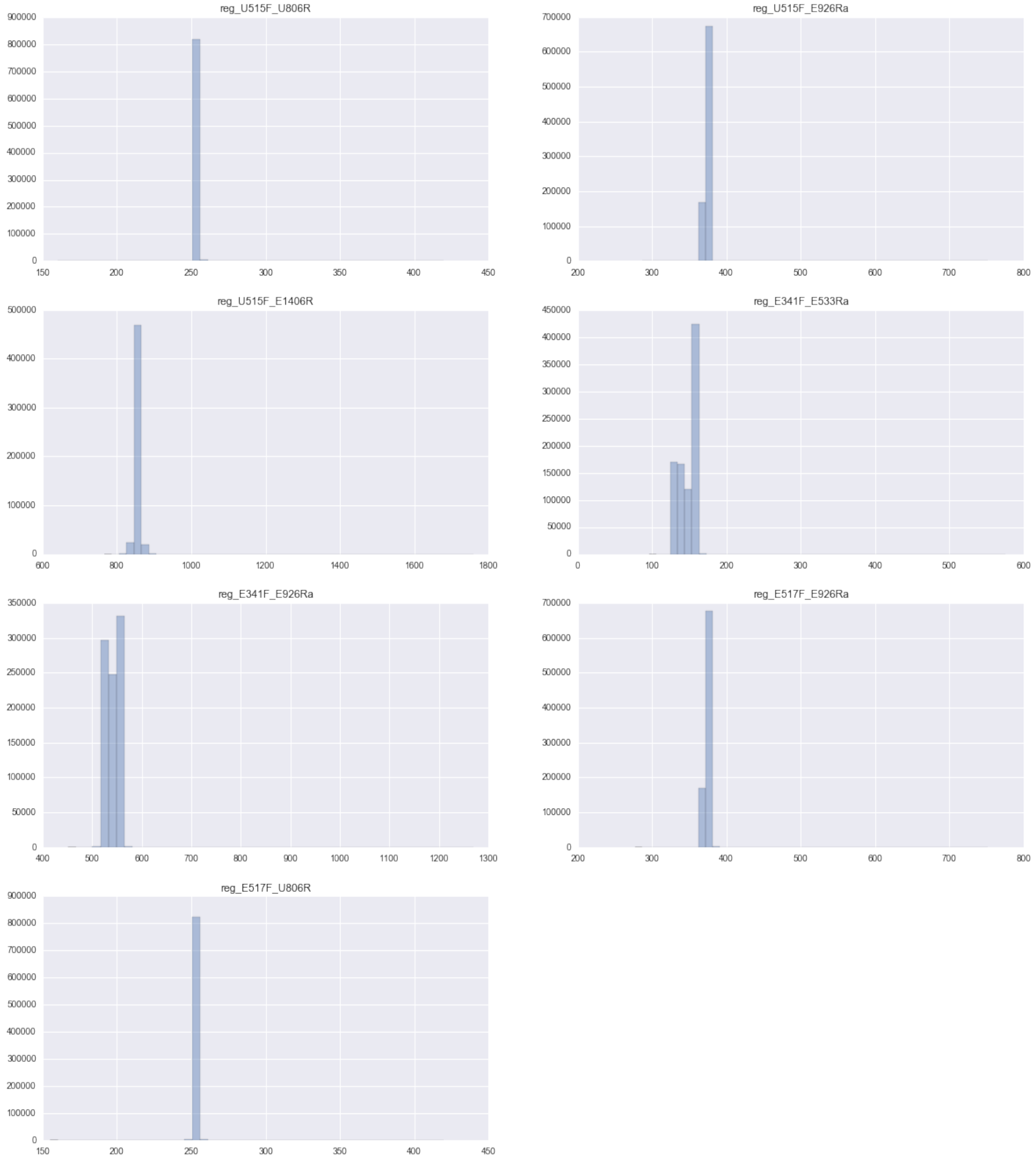

Figure 5: Distribution of the number of sequences (Y-axis) by length (X-axis) for the regions flanked by the selected primer pairs. *First row.*- U515F-U806R (left), U515F-E926Ra (right). *Second row.*- U515F-E1406R (left), E341F-E533Ra (right). *Third row.*- E341F-E926Ra (left), E517F-E926Ra (right). *Fourth row.*- E517F-U806R (left).

3.4.2. Sequence preprocessing

Having identified our regions of interest, and after counting the number of chosen sequences grouped per phylum, we decided finally to put apart only the sequences belonging to groups with more than 100 sequences. Sequences belonging to phyla with less than 100 sequences each, were not taken into account because we wanted to have test sets with at less 30 sequences per phylum. This gave us a range of 365 different taxonomies to assign, each of them up to genus level.

When a set of DNA sequences is going to be used whereas to train the classifier or to use it to predict their taxonomic assignment, it is necessary to turn it into an appropriate numerical form before. The path to follow is drawn in the algorithm [2], that seizes Fourier Analysis to get the power spectra from the sequences, previously mapped onto a $\mathbb{R}^3$ space yielded by their tetrahedron vertex projections.

3.4.3. Out-of-Bag error

Although a typical *Random Forests* implementation requires of fixing a number of diverse parameters in order to perform a supervised training, two of them i.e. the number of of trees or estimators, and the size of the feature subspace, were fixed through the analysis of the *out-of-bag* error. In order to understand this type of error, it is necessary to consider that every time a tree is built, a random sampling with replacement (bootstrapping) on the feature space is done.

So, at any time we will always have a bootstrapped data set (used by the tree) and a set of *out-of-bag* elements that were not taken by the sampling. Out-of-bag estimate for the generalization error is the error rate of the out-of-

Algorithm 2: Sequence preprocessing.

```
input  : A set $S$ of $p$ DNA sequences
output: A features matrix $X_{p \times m}$ with $X[i, j] \in \mathbb{R}$
begin
    $PS \leftarrow$ an empty list
    for $i \leftarrow 1$ to $p$ do
        $\chi_{l \in \{r,g,b\}} \leftarrow$ tetrahedron_mapping($S[i]$)
        $F_i[n] \leftarrow \sum_{n=0}^{N-1} \chi_l[n]\mathbf{w}^{-n},\ l \in \{r, g, b\}$
        $PS[i] \leftarrow \sum_{l \in \{r,g,b\}} |F_i[l]|^2$
        if $n$ is even then
            $t \leftarrow int(n/2)$
        else
            $t \leftarrow int(n/2) + 1$
        $PS[i] \leftarrow PS[i][1..t]$
    $m \leftarrow$ max length $PS$
    for $i \leftarrow 1$ to $p$ do
        $n \leftarrow$ length $PS[i]$
        if $n < m$ then
            $PS[i] \leftarrow$ akima_interpolation($PS[i], m$)
    $X \leftarrow$ zeros($p, m$)
    for $i \leftarrow 1$ to $p$ do
        for $j \leftarrow 1$ to $m$ do
            $X[i, j] \leftarrow PS[i, j]$
    $X \leftarrow$ normalize($X$)
```

bag classifier on the training set. Using a small subset of mapped sequences (500 in our case) and varying the number of estimator (from 90 to 400) fig. 6 points out this error rate, regarding to three different strategies to define the size of the sampling space.

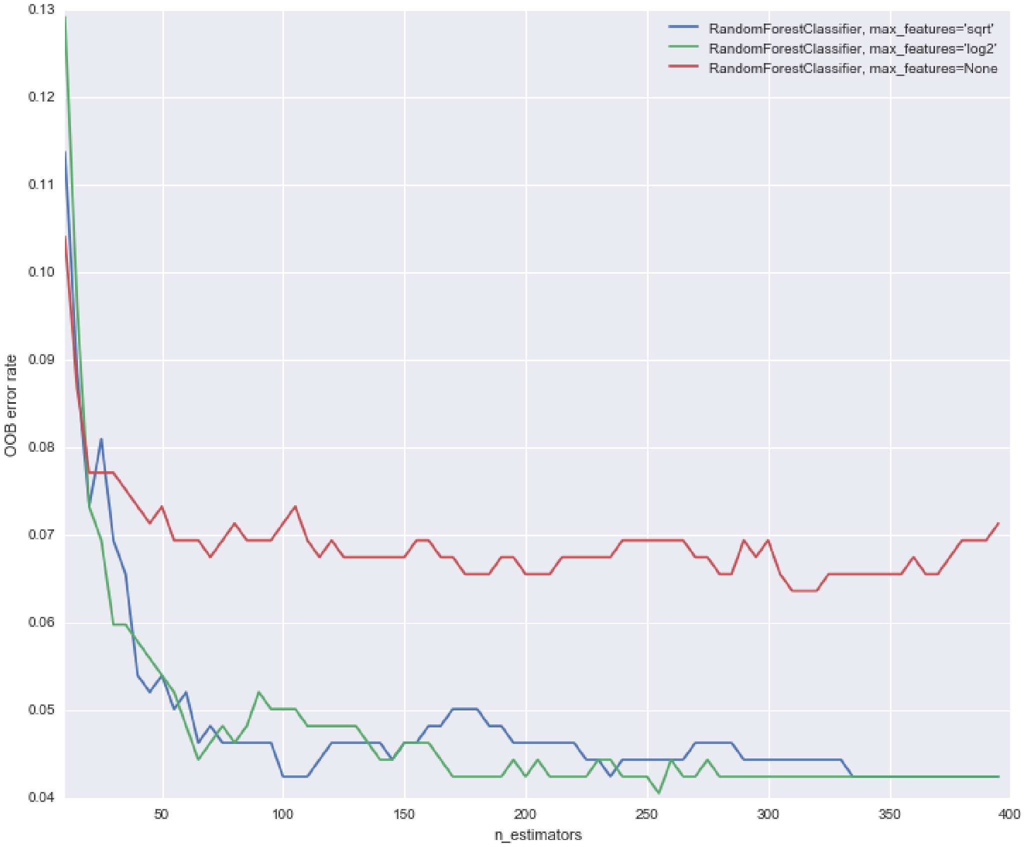

Figure 6: Out-of-bag error measured at the addition of each new tree during training, for three sample sizes: $log(n)$, $sqrt(n)$ and n (number of features in the whole training set). The number of estimators is displayed at the X-axis, and the error rate at the Y-axis.

In consequence, due to it is not appreciable a significant variation in the error rate when the number of estimators is 100 or more, that value (100) was used to train our model. Due to both $sqrt(n)$ and $log(n)$ have lower rates than n (here, n is the number of features in a training sample), and are preferable to the last one to set the sample size, this was set to $sqrt(n)$ in our algorithm. The above is because we wanted to have the simplest model possible to contrast with other well established methods to assign taxonomy.

3.4.4. Setting up the classifier

Once the number of estimators, and the sampling size were defined, we went ahead with the training of a *Random Forests* classifier. This process would depend on the implementation being used. Nevertheless, since the DNA sequences are not evenly distributed among taxa at different levels, the original set of preprocessed features X has to be splitted in a stratified fashion. Seeking to put aside 75% of the input features for classification and the remaining 25% for testing, the division was done by enforcing this proportion between the sequences grouped by either phylum or genus.

But, there will be taxa constituted by a very small set of sequences in comparison to other ones with thousands of sequences, giving place to an unbalanced classification problem. To mitigate this situation, we associated weights to the labels before the classifier starts to be trained. Those weights are calculated from the values of y and are inversely proportional to class frequencies. The vector y contains a finite set of numbers in $\mathbb{N}$, where each number identifies a specific taxon.

3.4.5. Assigning taxonomy

The classifier that we have built, takes a set of DNA sequences flanked by a primer pair known in advance, and assigns a taxonomy to them. In this work, we have restricted our classification space to phylum taxa with at least 100 sequences each, and flanked regions of less than 280bp. Therefore, the number of bacterial sequences annotated up to genus level matched by the primer pair E517F-U806R, was slightly reduced from 542.808 to 519.129. Besides, as the length of these regions was limited to 280bp, during their preprocessing sequences were scaled up to 140 elements in frequency domain. It is worth to recall that power spectrum has one half of length of the original signal (with the other half portraying the complex conjugate of the former).

There are 365 different assigned taxonomies up to genus level in the filtered data set, or what is the same, there are 365 different labels or classes to train a classifier. It would be computationally expensive to train a classifier under these constraints, though it also would tend to bring out inaccurate predictions due to the range of variations in terms of number of sequences per label. We tackle this problem in a two-stages approach.

At first, we trained a classifier CLF_1 using the whole set of processed sequences, in such a way that it could recognize the phylum they belong to. Looking at the table 2, it is noticeable that roughly 94% of the 519.129 sequences are distributed between just 4 phyla, whereas the remaining 27.350 are assigned to 13 phyla. Clearly, we are in front of an unbalanced classifying problem, so it was imperative to assign a weight to each phylum's label during the training of the *Random Forests* model.

In the second stage and once we know the phylum of a sequence, the next step is to do its taxonomic assignation. For that reason, the whole data set has to be filtered by the assigned phylum so we can use a reduced set to train another classifier $CLF_2[i]$ where $i \in [1..17]$ and with the same

Phylum	Total Seq	Total Genera
Firmicutes	226277	90
Proteobacteria	128295	162
Actinobacteria	73670	39
Bacteroidetes	63537	31
Cyanobacteria	7745	10
Fusobacteria	7444	3
Spirochaetes	3622	3
Verrucomicrobia	2763	7
Thermi	1151	4
Tenericutes	1119	2
Planctomycetes	821	3
Acidobacteria	706	2
Synergistetes	703	3
Nitrospirae	669	2
SAR406	287	2
Thermotogae	182	1
Deferribacteres	138	1

Table 2: Phyla assigned to regions matched by E517F-U806R and filtered by length and minimum number of occurrences. The first column (Phylum) contains the phylum's name, the second one (Total Seq) has the total amount of sequences in each phylum, and the third column (Total Genera) contains the different number of genera per phylum.

parameters than CLF_1, but with the aim of defining which of the phylum's genera the sequence belongs to. Practically, if we extend this approach, it will be necessary to train $N+1$ classifiers i.e. one (CLF_1) to assign a phylum from between the N phyla covered by the training set, and N classifiers $CLF_2[n]$ to assign taxonomy up to genus level depending on the genera present in the data set filtered by the n-th phylum. In the algorithm [3] we have devised, there are calls to routines that resembles to those present in the *Random Forests* implementation used in this research.

Algorithm 3: Assigning taxonomy.

```
input  : A set $S$ of $k$ DNA sequences
output: A labels vector $y$ with $y[i] \in [1..k]$
begin
    $X \leftarrow$ preprocess(A)
    $CLF_1 \leftarrow$ load phylum trained classifier
    $CLF_2 \leftarrow$ an empty associative array
    $y \leftarrow$ an empty list
    for $i \leftarrow 1$ to $k$ do
        $p \leftarrow CLF_1$.predict($X[i]$)
        if $p$ not in $CLF_2$.keys() then
            $CLF_2\{p\} \leftarrow$ load $p$-th trained classifier
        $y[i] \leftarrow CLF_2\{p\}$.predict($X[i]$)
```

The algorithms designed in this work were primarily coded with Python. All the involved numerical processing was done using NumPy [Walt et al., 2011], Pandas [McKinney, 2010] and SciPy Oliphant [2007] libraries. DNA sequence processing was done with the aid of Scikit-Bio [Scikit-Bio Development Team, 2015]. *Random Forests* classifiers were built using the Scikit-Learn [Pedregosa et al., 2011] implementation. Biopython [Cock et al., 2009] was used in the phylogenetic tree construction.

4. Results and Discussion

With the aim of verifying that the Fourier Analysis stated in this paper permits to measure similarity in 16S, a coding region from bacterial DNA, we arbitrarily selected the fifteen sequences in table 3 from three different phyla. Then, two dendrograms were generated using i) the algorithm [1] we propose (fig. 7); and ii) a neighbor-joining clustering using a pairwise distance matrix that was built from a multiple sequence alignment made with CLUSTAL [Sievers et al., 2011], a software based on a progressive alignment heuristic (fig. 8).

Both trees present a similar conformation, especially around grouping sequences that come from the same phylum. Even more, the region associated to *Lachnospiraceae Blautia* (294759) was positioned near sequences from *Bacteroidetes* phylum both by CLUSTAL and our method. Nevertheless, what we are looking for in constructing these trees is the suitability of a similarity measure given by the pairwise comparison of the power spectra from 16S rRNA regions in accordance to the way we propose to get a numerical representation of the DNA fragments, instead of develop an evolutionary explanation about discrepancies between them.

To assess the classification power of the devised algorithm, we proceeded with the taxonomy prediction of a test data set that were not used during the

Seq. id	Phylum	Family/Genus/Species
915470	Actinobacteria	Corynebacteriaceae; Corynebacterium; durum
925311	Actinobacteria	Micrococcaceae; Micrococcus; luteus
1085270	Actinobacteria	Micrococcaceae; Rothia; mucilaginosa
1016192	Actinobacteria	Micrococcaceae; Rothia; mucilaginosa
490191	Actinobacteria	Nocardiaceae; Rhodococcus; fascians
3270614	Bacteroidetes	Bacteroidaceae; Bacteroides; uniformis
653192	Bacteroidetes	Porphyromonadaceae; Porphyromonas; endodontalis
126842	Bacteroidetes	Prevotellaceae; Prevotella; copri
693510	Bacteroidetes	Prevotellaceae; Prevotella; melaninogenica
4431642	Bacteroidetes	Prevotellaceae; Prevotella; nigrescens
978699	Firmicutes	Staphylococcaceae; Staphylococcus; aureus
672096	Firmicutes	Staphylococcaceae; Staphylococcus; epidermidis
923503	Firmicutes	Staphylococcaceae; Staphylococcus; sciuri
294759	Firmicutes	Lachnospiraceae; Blautia; producta
541206	Firmicutes	Erysipelotrichaceae; Eubacterium; dolichum

Table 3: Fifteen DNA fragments compassed by E517F and U806R primers were used in the construction of the phylogenetic trees. The first column corresponds to the sequence identification in GreenGenes. The other columns contain the phylum and the taxonomy as from family level, respectively.

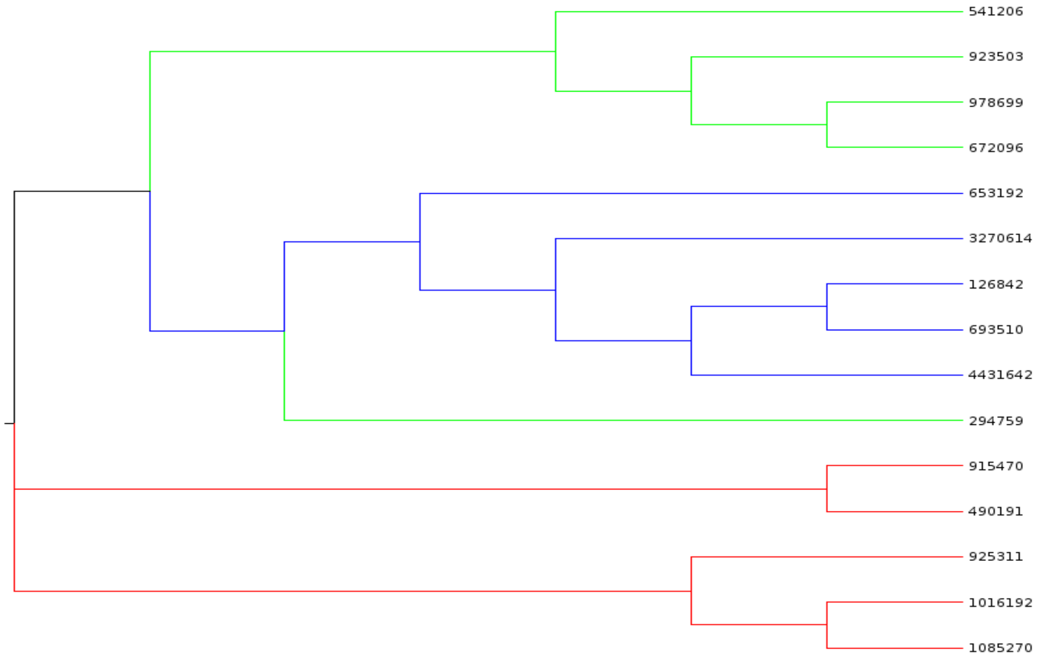

Figure 7: Dendrogram constructed through *neighbor-joining*, an agglomerative clustering method, as from the power spectra of each of the sampled sequences. Sequences identification numbers (seq. id) were written next to the leafs. Sequence's phylum is denoted by color: *Firmicutes* in green, *Bacteroidetes* in blue, and *Actinobacteria* in red.

training phase. Our classifiers operate in two stages. At the beginning, the space of possible taxonomies is reduced because the first classifier attempts to assign the data with a specific phylum, without having any ambiguity if it was possible. Accuracy of the classification at phylum level is explained by the confusion matrix in fig. 9. By definition, a confusion matrix C is such that $C_{i,j}$ is equal to the number of observations known to be in group i but predicted to be in group j [Pedregosa et al., 2011].

Precision and recall of the phylum classifier can be seen in table 4. The F1 score is reckoned at 0.98 with a precision of 100% which could be jus-

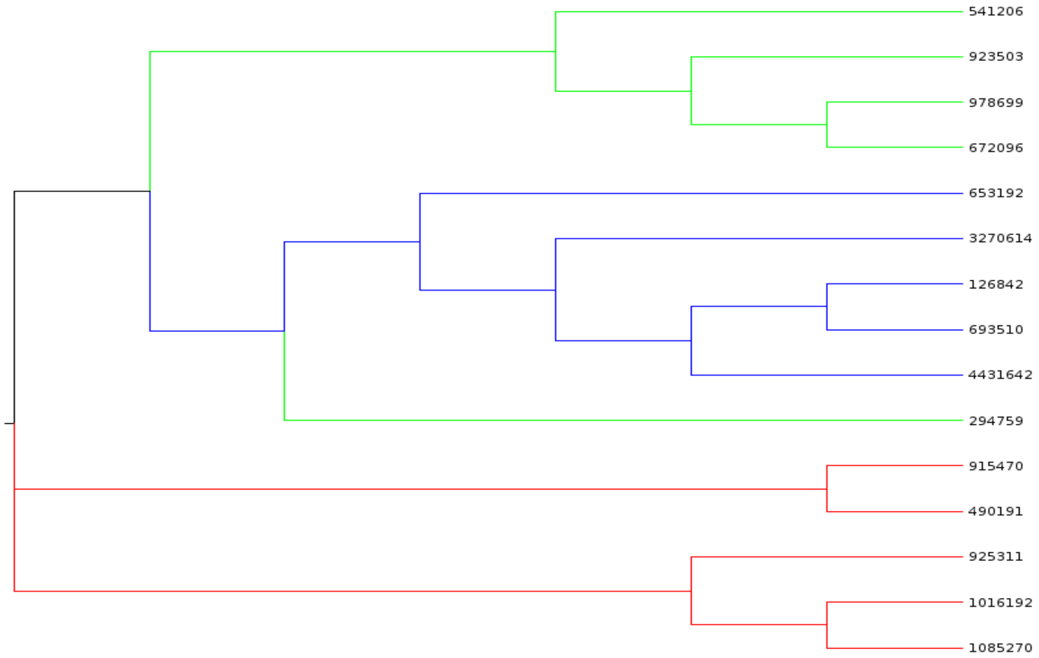

Figure 8: Dendrogram constructed as a result of a multiple sequence alignment with CLUSTAL. As in the other one, sequences identification numbers (seq. id) were written next to the leafs. Sequence's phylum is denoted by color: *Firmicutes* in green, *Bacteroidetes* in blue, and *Actinobacteria* in red.

tified as we have not trained the classifier with all the available sequences in GreenGenes, provided that they were matched by a primer pair. These ratios are defined next:

$$precision = tp/(tp + fp)$$

$$recall = tp/(tp + fn)$$

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

where tp is the number of true positives, fp is the number of false positives, and fn the number of false negatives.

As soon as a phylum is defined, another classifier trained to assign tax-

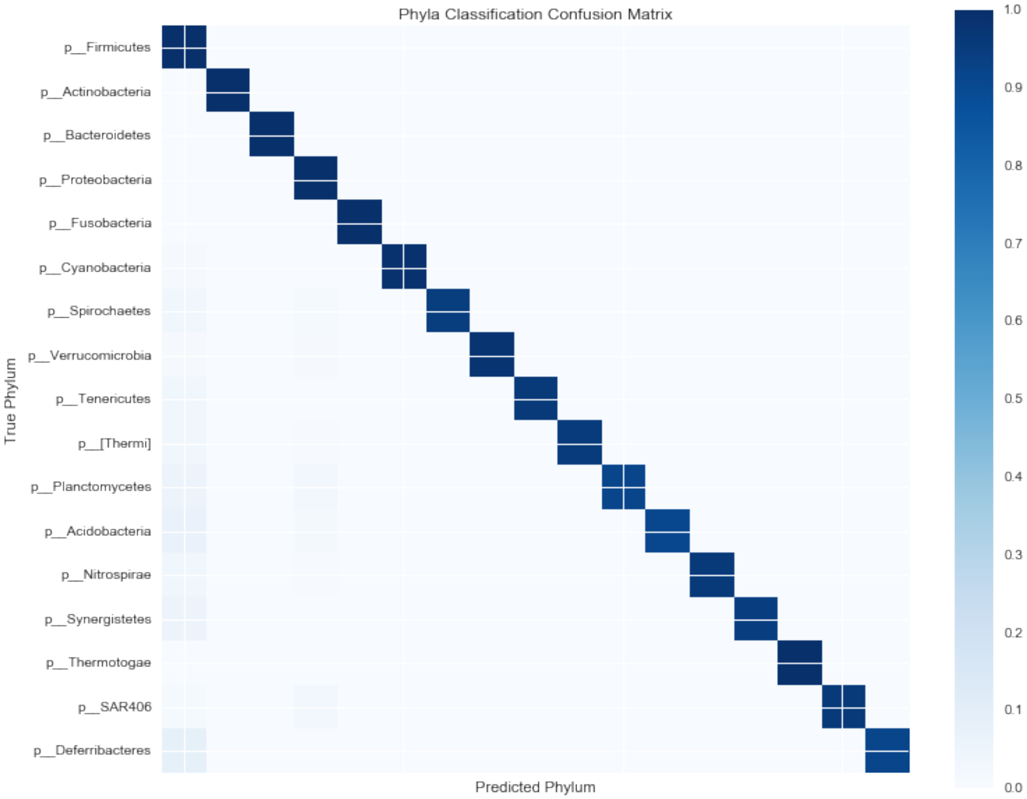

Figure 9: Confusion matrix for the phylum classifier. The phyla labels where positioned only in the Y-axis. For the X-axis, the labels are not shown. However, the top label in the Y-axis is equal to the label of the first square at X-axis, and so on. The cells are colored in function of how many of the true phyla (Y-axis) were actually predicted (X-axis). A depth blue color indicates that the phylum at the Y-axis was almost perfectly predicted by the classifier.

onomy up to genus level is enabled. For example, amid the phyla covered by the pair E517F-U806R, *Proteobacteria* phylum is divided into 160 different genera. Accuracy of the classification at genus level for the sample phylum is explained by the confusion matrix in fig. 10 as well as in the report given in table 5.

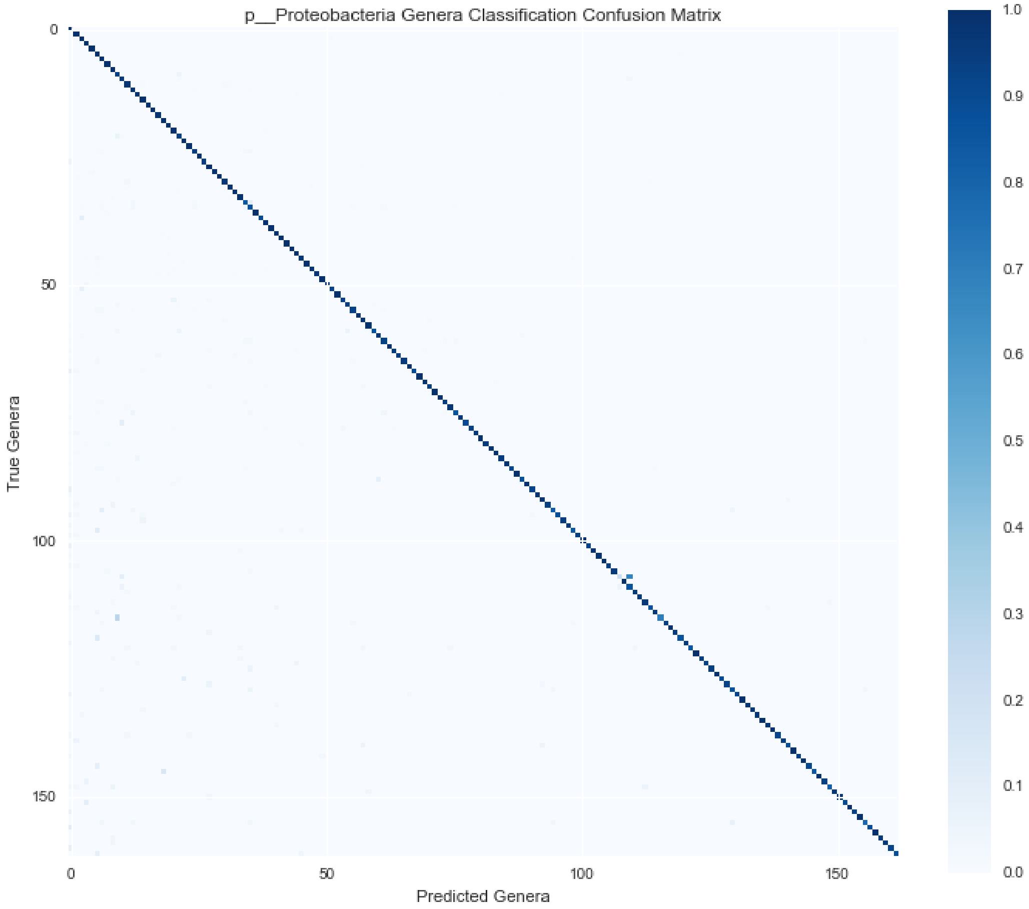

Figure 10: Confusion matrix for genus classifier within *Proteobacteria* phylum. The cells are colored in function of how many of the true genera (Y-axis) were actually predicted (X-axis). A depth blue color indicates that the genera at the Y-axis was almost perfectly predicted by the classifier.

Phylum	Precision	Recall	F1 Score	Support
Firmicutes	1	1	1	56570
Actinobacteria	1	1	1	18418
Bacteroidetes	1	1	1	15884
Proteobacteria	1	1	1	32074
Fusobacteria	1	1	1	1861
Cyanobacteria	1	0.99	0.99	1936
Spirochaetes	1	0.95	0.97	906
Verrucomicrobia	1	0.98	0.99	691
Tenericutes	1	0.96	0.98	280
Thermi	1	0.95	0.98	288
Planctomycetes	1	0.92	0.96	205
Acidobacteria	1	0.91	0.95	177
Nitrospirae	1	0.96	0.98	167
Synergistetes	1	0.94	0.97	176
Thermotogae	1	1	1	46
SAR406	1	0.96	0.98	72
Deferribacteres	1	0.91	0.96	35
Average/Total	1	0.97	0.98	129786

Table 4: Classification results for the first ensemble of random trees used to recognize which phylum a sequence belongs to, as follows: phylum's name (first column), precision (second column), recall (third column) and F1 score (fourth column). The last column displays the support, i.e. the number of occurrences of each class in the correct target values, for each phylum.

Genus	Precision	Recall	F1 Score	Support
Average/Total	0.98	0.95	0.96	32077

Table 5: Accuracy results for a classifier designed to identify and assign taxonomy up to genus level in *Proteobacteria* phylum.

Finally, we compared the proposed classifier with UCLUST [Ghodsi et al., 2011], RDP [Wang et al., 2007] and MOTHUR [Schloss et al., 2009], all of them working with the same set of 100 sequences, and using a reference dataset available at 97_otus.fasta.gz created by clustering all the sequences in the GreenGenes database into 97% identity clusters.

It is worth to say that, except by TAXOFOR, all the rest had between 83% (RDP) to 97% (UCLUST) of achievement in assigning taxonomy to each of the DNA sequences (which once again corresponded to regions flanked by E517F and U806R). Moreover, it is important to mention here that given that those programs have their own custom databases it is quite likely that they have been trained already with representatives of the test sequences, while TAXOFOR was tested including sequences it has not been trained on. At the same time, we decided to analyze the performance of all the 4 tools, measured in CPU time, i.e. without considering I/O disk operations. According to table 6, and without considering implementation details of each of the other programs that can affect their performance by differences in the number of required computations to achieve their results, our classification schema is twice as fast as UCLUST, with a better level of accuracy, and considerable faster than both RDP and MOTHUR.

A major aim of this work was to verify the feasibility of Fourier Analysis

Program	CPU Time (s)	Precision
TAXOFOR	**21.874**	**1.00**
UCLUST	37.810	0.96
RDP	270.493	0.83
MOTHUR	601.258	0.97

Table 6: Comparison between TAXOFOR, UCLUST, RDP and MOTHUR in terms of CPU time and precision. CPU time in seconds (second column) was computed by adding up the user and kernel process time gauged in the same machine, an Intel(R) Xeon(R) CPU E5-2670 v2 at 2.50GHz with 3.75GB in RAM and Ubuntu 14.04. Precision is given in the third column.

in assigning taxonomy to 16S rRNA amplicons. We have elucidated a form of representing 16S rRNA genes numerically in such a way that it preserves the maximum amount of mutual and structural information [Leito et al., 2005], in contrast with other methods which are based on feature extraction (e.g. k-mer counting). We can shift from a time (or space) domain into a frequency domain through the application of a Discrete Fourier Transform to the numerical version of these regions, and use this transformed signals to simplify the training and application of a machine learning classifier.

After exploring a reference database like GreenGenes, is easy to recognize that we are in front of a training set where there is a big variation in the number of samples per class or ”label” to avoid the confusion with the homonymous taxonomic level. Certainly, the changes are so unexpected that phyla like *Firmicutes* can come to have hundreds of thousands of annotated sequences, whereas a phylum like *Chlamydiae* only has 358 sequences, many of them with incomplete taxonomy. Even worse, this number could be dramatically reduced when it comes into scene a given primer pair. For example, and following with *Chlamydiae*, one of our best primer pairs and eventually the most widely used here, i.e. E517F-U806R, matched only one of sequences from this phylum. A superficial analysis of fig. 3 led us to consider, once again, that those primer pair considered as universal are not so much.

5. Conclusions

Fourier analysis has shown to be a useful resource in order to get an efficient way to assign 16S rRNA sequences flanked by forward and reverse primers typically used in microbial surveys. In fact, using an ensemble of randomized trees as classifier, we have outperformed in terms of processing time three of the most popular available tools to do the same task. In addition to that, our classifier has proved to have an impressive prediction power with an average precision score of near 98% for the inspected taxa up to genus level, even without being trained with the whole set of sequences in GreenGenes database. We believed that there is enough room to improve our algorithm in a future version and release it to the community, considering that the software was written without any kind of optimization, in spite of having code structures susceptible to be parallelizable.

References

Akima, H., 1970. A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures. Journal of the Association for Computing

Machinery 17 (4), 589–602.

Anastassiou, D., Jul 2001. Genomic signal processing. IEEE Signal Processing Magazine 18 (4), 8–20.

Breiman, L., 2001. Random forests. Machine learning 45 (1), 5–32.

Caporaso, J. G., Lauber, C. L., Walters, W. a., Berg-Lyons, D., Huntley, J., Fierer, N., Owens, S. M., Betley, J., Fraser, L., Bauer, M., Gormley, N., Gilbert, J. a., Smith, G., Knight, R., 2012. Ultra-high-throughput microbial community analysis on the Illumina HiSeq and MiSeq platforms. The ISME Journal 6 (8), 1621–1624.

Chakravorty, S., Helb, D., Burday, M., Connell, N., 2007. A detailed analysis of 16S ribosomal RNA gene segments for the diagnosis of pathogenic bacteria. J Microbiol Methods 69 (2), 330–339.

Chenna, R., Sugawara, H., Koike, T., Lopez, R., Gibson, T. J., Higgins, D. G., Thompson, J. D., 2003. Multiple sequence alignment with the Clustal series of programs. Nucleic Acids Research 31 (13), 3497–3500.

Cock, P. J. A., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., De Hoon, M. J. L., 2009. Biopython: Freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics 25 (11), 1422–1423.

Coward, E., 1997. Equivalence of two Fourier methods for biological sequences. DNA Sequence, 64–70.

DeSantis, T. Z., Hugenholtz, P., Larsen, N., Rojas, M., Brodie, E. L., Keller, K., Huber, T., Dalevi, D., Hu, P., Andersen, G. L., 2006. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. Applied and Environmental Microbiology 72 (7), 5069–5072.

Ghodsi, M., Liu, B., Pop, M., 2011. DNACLUST: accurate and efficient clustering of phylogenetic marker genes. BMC Bioinformatics 12, 271.

Grant, W., Long, P., 1981. Environmental microbiology. Humana Press.

King, B. R., Aburdene, M., Thompson, A., Warres, Z., 2014. Application of discrete Fourier inter-coefficient difference for assessing genetic sequence similarity. EURASIP journal on bioinformatics & systems biology 2014 (1), 8.

Kwan, H. K., Arniker, S. B., 2009. Numerical representation of DNA sequences. Proceedings of 2009 IEEE International Conference on Electro/Information Technology, EIT 2009, 307–310.

Lane, D. J., Pace, B., Olsen, G. J., Stahl, D. A., Sogin, M. L., Pace, N. R., 1985a. Rapid determination of 16S ribosomal RNA sequences for phylogenetic analyses. Proceedings of the National Academy of Sciences 82 (20), 6955–6959.

Lane, D. J., Pace, B., Olsen, G. J., Stahl, D. A., Sogin, M. L., Pace, N. R., 1985b. Rapid determination of 16s ribosomal rna sequences for phylogenetic analyses. Proceedings of the National Academy of Sciences 82 (20), 6955–6959.

Leito, H. C. G., Pessa, L. S., Stolfi, J., 2005. Mutual information content of homologous DNA sequences. Genetics and molecular research : GMR.

Lilit Garibyan, N. A., 2014. Research Techniques Made Simple : Polymerase Chain Reaction (PCR). NIH Public Access 133 (3), 1–8.

Liu, Z., Desantis, T. Z., Andersen, G. L., Knight, R., 2008. Accurate taxonomy assignments from 16S rRNA sequences produced by highly parallel pyrosequencers. Nucleic Acids Research 36 (18), 1–11.

McKinney, W., 2010. Data structures for statistical computing in python. In: van der Walt, S., Millman, J. (Eds.), Proceedings of the 9th Python in Science Conference. pp. 51–56.

Oliphant, T. E., 2007. Python for scientific computing. Computing in Science & Engineering 9 (3), 10–20.

Orfanidis, S. J., 2009. Introduction to Signal Processing. Prentice Hall Inc.

Pace, N. R., 1997. A molecular view of microbial diversity and the biosphere. Science 276 (5313), 734–740.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Rafiei, D., Mendelzon, A., 1998. Efficient Retrieval of Similar Time Sequences

Using DFT. In: The 5th International Conference on Foundations of Data Organization. p. 0.

Saitou N, N. M., 1987. The Neighbor-joining Method: A New Method for Reconstructing Phylogenetic Trees. Mol. Biol. 4 (4), 406–425.

Sanschagrin, S., Yergeau, E., 2014. Next-generation sequencing of 16S ribosomal RNA gene amplicons. Journal of visualized experiments : JoVE - (90), 1–7.

Schloss, P. D., 2010. The Effects of Alignment Quality , Distance Calculation Method , Sequence Filtering , and Region on the Analysis of 16S rRNA Gene-Based Studies. PloS Computational Biology 6 (7).

Schloss, P. D., Westcott, S. L., 2011. Assessing and improving methods used in operational taxonomic unit-based approaches for 16S rRNA gene sequence analysis. Applied and Environmental Microbiology 77 (10), 3219–3226.

Schloss, P. D., Westcott, S. L., Ryabin, T., Hall, J. R., Hartmann, M., Hollister, E. B., Lesniewski, R. A., Oakley, B. B., Parks, D. H., Robinson, C. J., Sahl, J. W., Stres, B., Thallinger, G. G., Van Horn, D. J., Weber, C. F., dec 2009. Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. Applied and environmental microbiology 75 (23), 7537–41.

Scikit-Bio Development Team, 2015. scikit-bio.
URL http://scikit-bio.org

Sievers, F., Wilm, A., Dineen, D., Gibson, T. J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M., Söding, J., Thompson, J. D., Higgins, D. G., 2011. Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. Molecular Systems Biology 7 (1).

Silverman, B. D., Linsker, R., 1986. A measure of DNA periodicity. Journal of Theoretical Biology 118, 295–300.

Soergel, D. a. W., Dey, N., Knight, R., Brenner, S. E., 2012. Selection of primers for optimal taxonomic classification of environmental 16S rRNA gene sequences. The ISME Journal 6 (7), 1440–1444.

Teyssier, C., Ramuz, M., Jumas-bilak, E., 2003. Atypical 16S rRNA Gene Copies in. Society 185 (9), 2901–2909.

Voss, R. F., 1992. Evolution of long-range fractal correlations and 1/f noise in DNA base sequences. Physical Review Letters 68 (25), 3805–3808.

Walt, S. v. d., Colbert, S. C., Varoquaux, G., 2011. The numpy array: A structure for efficient numerical computation. Computing in Science & Engineering 13 (2), 22–30.

Wang, Q., Garrity, G. M., Tiedje, J. M., Cole, J. R., 2007. Naïve Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. Applied and Environmental Microbiology 73 (16), 5261–5267.

Woese, C. R., Fox, G. E., 1977. Phylogenetic structure of the prokaryotic domain: the primary kingdoms. Proceedings of the National Academy of Sciences of the United States of America 74 (11), 5088–5090.

Yin, C., Chen, Y., Yau, S. S.-T., 2014. A measure of DNA sequence similarity by Fourier Transform with applications on hierarchical clustering. Journal of theoretical biology 359C, 18–28.

Yin, C., Wang, J., 2014. A Novel Method for Comparative Analysis of DNA Sequences by Ramanujan-Fourier Transform. Journal of computational biology : a journal of computational molecular cell biology 21 (12), 1–26.

Yin, C., Yau, S. S.-T., 2015. An improved model for whole genome phylogenetic analysis by Fourier transform. Journal of Theoretical Biology 382, 99–110.