# NOVEL METHODS FOR THE COMPUTATIONAL ANALYSIS OF CODON USAGE BIAS

A THESIS SUBMITTED TO
THE UNIVERSITY OF KENT
IN THE SUBJECT OF COMPUTER SCIENCE
FOR THE DEGREE
OF PHD.

By
Yun Deng
June 2019

# Abstract

The genetic code encodes the same amino acid with multiple codon choices, but in a biased fashion. This phenomenon is called the codon usage bias (CUB). There have been significant research efforts trying to quantify codon usage bias and probe into its origins. Understanding CUB is important for at least two reasons. Firstly, it is connected with gene expression, and thus of fundamental importance for our understanding of life. Secondly it is important for the optimisation of heterologous gene expression in industrial bioproduction including the pharmaceutical industry. This thesis makes three main contributions to the understanding of CUB: (1) It proposes a novel measure of codon usage bias which does not require any context information other than the nature of the coding sequences themselves. The proposed measure is capable of quantifying codon usage bias at different levels of an individual sequence, a particular amino acid type, and a whole genome, and also capable to provide comprehensive and desired CUB information for the correlation study about specific CUB related factors by constructing high dimensional CUB feature spaces. (2) It derives a stochastic thermodynamic based model to investigate what the evolutionary drivers of codon usage bias are from a macroscopic perspective. (3) It applies the proposed methods to extensive genomic data. Our main conclusions derived from the applications to real organisms include (a) codon usage bias and gene lengths cooperate together to satisfy different protein requirements in the cells; (b) codon usage bias correlates with phylogenetic distances among remote groups of species; (c) codon usage bias cannot be explained solely by selection pressures that act on the genome-wide codon frequencies, but also includes pressures that act at the level of individual genes.

# Acknowledgements

I am profoundly grateful to my supervisor Dominique Chu, without whom my research even the continuation of my career would have been impossible, and who lead me making forward step by step so magically that I frequently surprised myself with my improved research skills without feeling stress, and who raised me up to a brighter career path with his smart ideas and efficient supervisions.

Heartfelt thanks to my second supervisor Tobias von der Haar, who consistently offered powerful support and great patience like sunshine leading me through darkness, and who is like an encyclopedia I can refer to for wise and quick resolutions whenever my thoughts tangled into a mess.

My sincere thanks to Jeremie Kalfon, my team member, who showed great talent and offered me good resources to broaden my view and improve my research skill.

Last but not least, big thanks to my parents and my husband who supported me with their unconditional love. Big thanks to my mother-in-law who rescued me at the time I almost drowned in the cry of two little babies. And big thanks to those two crying little babies who had grown up to sweet big boys when I was writing up this thesis, and who never love me less even I often scolded them to go away when I need concentration, and who dug out cereals and filled their tummies but still hugged me happily when I missed their super time.

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

During the expression from genes to proteins, most amino acids are encoded by more than one nucleotide triplet or codon. Despite encoding the same amino acid, such synonymous codons are not biologically equivalent and there are usually large differences in their usage frequency. This phenomenon is called codon usage bias (CUB). Understanding CUB is the main topic of this work.

CUB has been attracting much research interest, and there are 3240 articles in 2018 alone according to Google Scholar. CUB has fundamental significance for understanding the principles of protein synthesis in biology, and is practically important in the context of heterologous gene expression in industrial bioproduction including pharmaceutical industry.

Research into CUB is mainly divided into two classes, one is construction and application of algorithms to quantify and also simulate CUB, the other is the investigation of the origins of CUB.

Existing CUB measures often require external reference information besides gene sequences such as tRNA abundances.This makes these measures difficult to use especially for poorly characterised organisms, which are the majority. There is thus a need for new measures that do not require any reference information.

Among the studies for understanding the evolutionary origins of codon usage bias, many putative evolutionary drivers for codon usage bias have been proposed, which involve broad range of factors from intrinsic features of genomes to elements playing key roles at different stages of the gene expression procedure. However as of yet there is no agreement in the community as to how to combine all the CUB driving forces together in a reasonable way to analyse CUB, and there may be many undiscovered drivers still awaiting discovery.

In our work we made the following contributions:

1. We propose a novel set of CUB measures:

   (a) $Sn$ measures CUB for a particular amino acid type in a specific sequence. It reflects the strength of forces to drive a codon sequence deviating from the state of random codon usage.

   (b) $MD$ measures CUB for a particular amino acid type at the whole genome level. It combines $Sn$ values for a type of amino acid in all the genes within the genome.

   (c) $\mathcal{MD}$ is a genome wide CUB measure which combines CUB information for all the genes and all the amino acids throughout the genome. It takes the form of a vector containing $MD$ values of all the amino acids.

2. We apply the proposed measures to fungal species and have the following findings:

   (a) CUB and gene length cooperate to satisfy the demand of protein production in cells.

   (b) Sequence specific CUB patterns among orthologs correlate with gene functions.

   (c) Amino acid specific CUB patterns correlate to amino acid chemistry.

   (d) Genome wide CUB patterns correlate to phylogenetic distances between species.

3. We propose a novel model based on concepts from statistical mechanics to explore CUB origins.

4. We apply the proposed CUB model to real genomes of three kingdoms (fungi, bacteria and protist) and find that there must be significant selection pressures on codon usage bias at the level of individual gene sequences. Our SLS model captures not only the genome-wide frequency features of codon usage but also the distribution of CUB across the genome.

The structure of this thesis is shown as follows:

1. Chapter **Introduction**: introduces the main topic of this work, and demonstrates the significance of this work by summarising previous research in that field.

2. Chapter **Literature Review**: summarises current available CUB measures and assumptions of CUB origins in a systematic way.

3. Chapter **A Novel Codon Usage Bias Measure**: describes the concepts and algorithms which our sequence specific and amino acid specific CUB measure $Sn$ is built on, also its application to fungal species regarding correlation study between protein abundance and CUB, and CUB patterns in homologous genes.

4. Chapter **Genome Wide Codon Usage Bias Analysis**: describes the concepts and algorithms which our genome wide CUB measures $MD$ and $\mathcal{MD}$ are based on, also its application to fungal species assisted by machine learning techniques.

5. Chapter **Stochastic Thermodynamics Based Model to Simulate Genome-wide CUB Pattern**: describes the concepts and algorithms which our 'Beanbag Model' and 'Sequence Selection Model' are based on, and their application to investigate whether there exists sequence level selection in species across fungi, bacteria and protist kingdoms.

6. Chapter **Conclusion**: summaries our contributions.

# Chapter 2

# Literature Review

To better understand codon usage bias, the main topic of this work, we first review relevant biological background.

## 2.1 Understanding Protein Synthesis Steps

Protein synthesis is a process whereby the genetic information is decoded from genome to proteome. Coding regions (exons) of genome are important for generating protein sequences. Noncoding regions of genome contain regulatory regions, introns (transcribed into mRNA but not expressed into protein), repetitive DNAs. Noncoding regions take up various proportions in eukaryotic genomes for example 98% in human genome, and almost 25% of the yeast genome (Parker et al. (2018)).

There are two key stages of protein synthesis: transcription and translation. In prokaryotes transcription and translation occur almost simultaneously on the freely floating DNAs in the cell cytoplasm. Here we mainly discuss eukaryotic protein synthesis.

### 2.1.1 Transcription

Transcription is a process whereby a mRNA chain is generated based on a DNA template. The DNA double helix in the genome is unzipped, and one strand acts as the template for RNA synthesis. This process is regulated by transcription factors (TF) and coactivators, where nucleoside triphosphates (NTPs) serve as the mRNA building materials and provide energy (NTPs to build mRNA include ATP, GTP, CTP and UTP). A schematic diagram briefly demonstrates transcription in Figure 1.

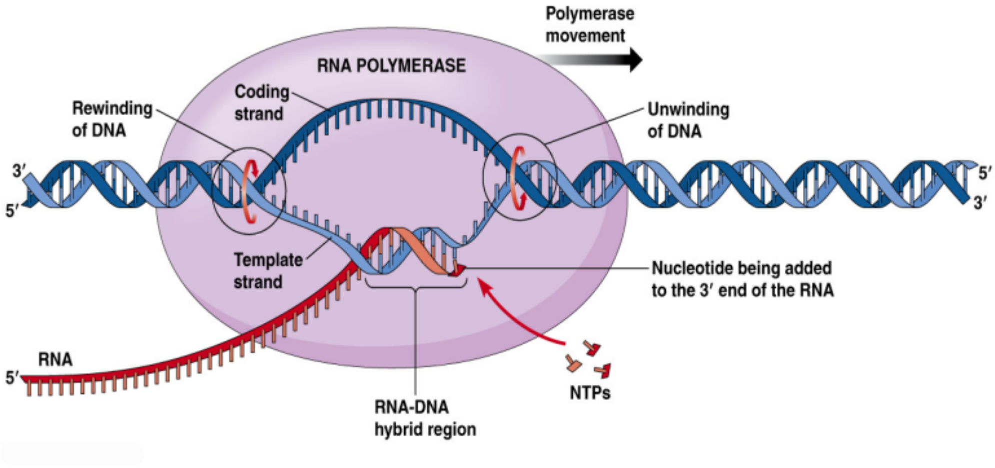


Figure 1: Transcription, diagram reference: Principles of cell biology (BIOL2060): Extracellular structures

mRNA is a copy of DNA genetic information, and then works as a template for assembling amino acids. mRNA is a single sequence of ribonucleotides. Ribonucleotide is composed of phosphate, ribose and base (include adenine [A], guanine [G], cytosine [C], uracil [U]). Each 3' carbon atom of ribonucleotide connects to 5' carbon atom of adjacent ribonucleotide by phosphorus ester bond to form the final mRNA. The two ends of the final mRNA are called 3' (3-prime, with a free hydroxyl group) and 5' (5-prime, with a free phosphate group). mRNA primary structure is shown in Figure 2.

Primary structure of DNA is very similar to mRNA but differs in two aspects: using deoxyribose rather than ribose (no hydroxyl attaches to 2' carbon in deoxyribose); and using base pair thymine [T] rather than uracil [U].

Transcription proceeds in the following general steps:

(1) Initiation: RNA polymerase is the main transcription enzyme during transcription. RNA polymerase together with necessary transcription factors (TF) correctly identify and combine to the specific sequence (called 'promoter') on a DNA template.

(2) Elongation: RNA polymerase "walks" along one strand of DNA as the template while new RNA strand is built. DNA double helix sequentially open

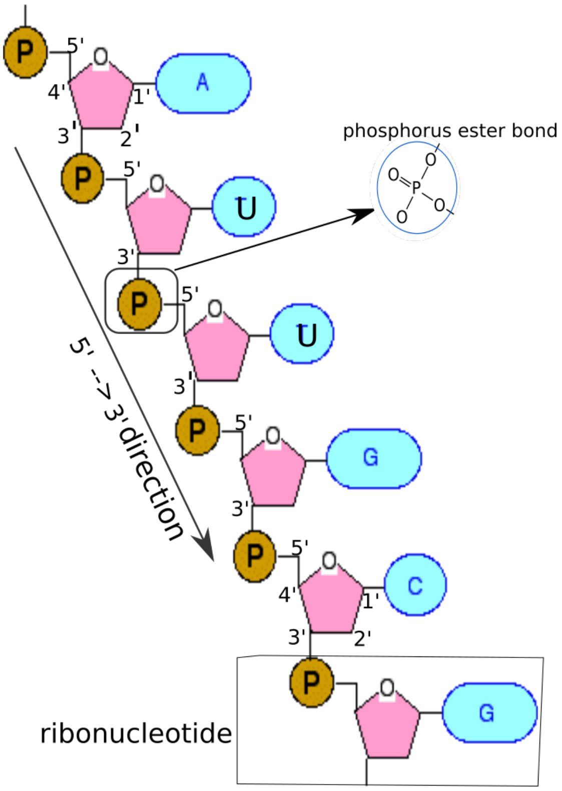


Figure 2: mRNA structure

to accept the new complementary mRNA base pairing and then re-close back to the original double-stranded structure. General RNA synthesis rate is 25 to 50 nucleotides/sec in prokaryotic, compared to 45 to 100 nucleotides/sec in eukaryotic (Uzman (2003)).

(3)Termination: Transcription terminates when RNA polymerase gets the stop signal (terminator) and detaches from template DNA while synthetic RNA are released. In prokaryotes, the terminator usually ends with a specific termination sequence which assists mRNA forming a G-C-rich hairpin loop and then causing polymerase to stall. Eukaryotic genes have the terminator as a sequence with some specific patterns at the 3' end. Such patterns are rich in AT (AATAA (A) or ATTAA (A), etc) and followed by TTTT (usually 3 to 5 T) at the distance of 0 ˜ 30 base pairs away (Lykke-Andersen and Jensen (2007)).

(4)Post-transcription modification: In eukaryotes, one more step is required to achieve functional mRNA. Primary eukaryotic mRNA (transcript precursor) eventually processes into the mature mRNA in four steps (1) Adding methylation cap at the 5' end. (2) Adding poly A tail at the 3' end. (3) Introns are spliced and exons are connected. Exons contain specific sequences called exonic splicing enhancers and exonic splicing silencers in order to enhance or depress splicing at the splice site. (4) Some parts of the mRNA are methylated.

### 2.1.2 Translation

Translation produces proteins using mRNAs as templates. One or more polypeptides constitute the protein and a polypeptide is a string of amino acids. Adjacent amino acids connect to each other by the covalent chemical bond formed between the carboxyl group of one amino acid and the amino group of the other amino acid.

The mRNA is read in its 5' to 3' direction, meanwhile the encoded polypeptide is made from its amino end (N-terminus) towards its carboxyl end (C-terminus). mRNAs, ribosomes, tRNAs, amino acids are involved in translation. A schematic diagram briefly illustrates transcription as Figure 3

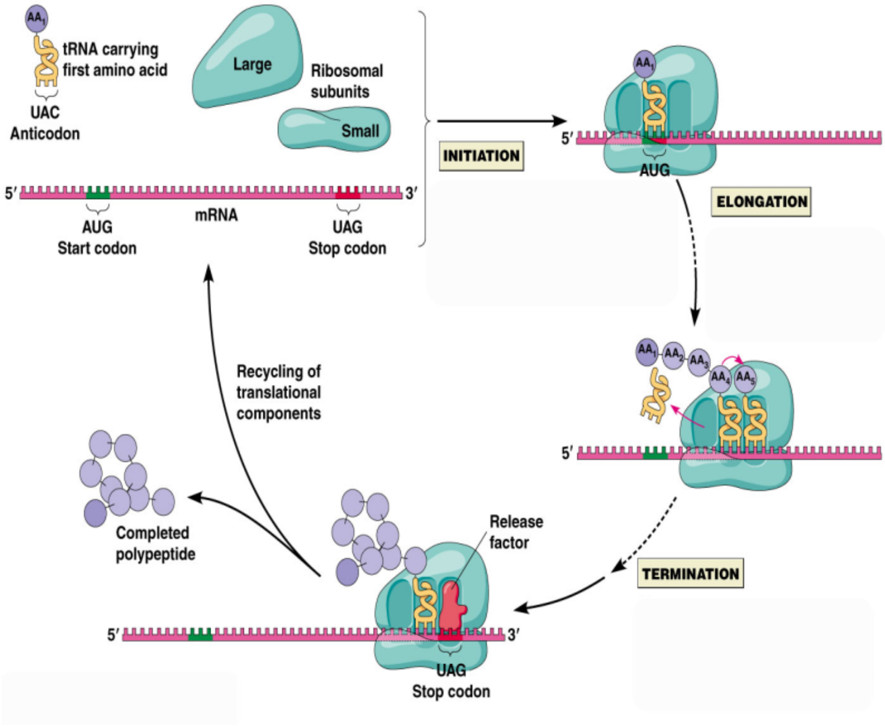


Figure 3: Translation, diagram reference: Principles of cell biology (BIOL2060): Extracellular structures.

A ribosome is composed of the ribosomal protein and the ribosomal RNA (rRNA), and it has 'A', 'P', 'E' three sites to accommodate tRNAs during translation.

The mRNA template is read by the unit of three adjacent nucleotides, which is called a codon. Genetic information carried by a mRNA is displayed as codon sequences. AUG is called the start codon where the translation starts, and UAA, UAG, UGA are called the stop codon where the translation terminates.

A tRNA is responsible for carrying an amino acid to the ribosome. Before interacting with the mRNA, the tRNA undergoes aminoacylation (a process covalently linking an amino acid to the 3' end of the tRNA) and becomes an aminoacyl-tRNA. A translation speed model states that global translation speed is enhanced by the amino acyl-tRNA competitors for ribosomes rather than tRNAs (Chu, Barnes and von der Haar (2011)).

Now we look into the process of translation in detail. Translation includes the following phases:

(1) Initiation: Initiation complex forms and scans the mRNA to locate the start codon. Initiation complex formation is a complicated procedure involving initiation factors, GTP, ribosomal small and large unites and the aminoacyl-tRNA which carries amino acid Met (Met-tRNA).

There are three initiation factors in prokaryotes, IF-1, IF-2 and IF-3, while eukaryotes have more. However the basic steps are the same: including formation of a ribosome small subunit (40S) initiation complex, finding the start codon (Met-tRNA matches the start codon at the P-site of the ribosome), and final formation of a big subunit (80S) initiation complex. The main difference between prokaryotes and eukaryotes lies in whether the small ribosomal subunit combines to the mRNA before it combines to the Met-tRNA (Berg, Tymoczko and Stryer (2002)).

When the first ribosome moves away from the start codon about 40 nucleotides, a second ribosome can attach to the start codon and begin another protein translation.

(2) Elongation: Ribosomes slide along the mRNA templates and connect the amino acids. The ribosome emerging from the initiation pathway undergoes a cyclical series of reactions. Each cycle includes the following three steps (Sesma and Von der Haar (2014)):

- A new aminoacyl-tRNA bind to A-site by hydrogen bonds according to codon and anti-codon pairing rules;
- The A-site tRNA takes over the peptide from the P-site tRNA, then the P-site tRNA moves to the E-site and the A-site tRNA to the P-site.
- The aminoacyl-tRNA loses amino acid and leaves the ribosome from E-site. The ribosome moves one codon distance along the mRNA towards the mRNA 3' end.

During elongation once the first two positions of codon are paired with anti-codon, exact base pairing of the third codon position is less critical. Such non-Watson-Crick base pairs without impacting aminoacyl-tRNA binding to mRNA is called 'wobble base pair' (Crick (1968)). Most organisms have fewer than 45 species of tRNAs (Chan and Lowe (2008)), and wobble pairings between the aminoacyl-tRNA and the mRNA guarantee all the codons can be recognised by available tRNAs. However tRNAs only wobble to match a codon if there is no better tRNA for that codon (Percudani, Pavesi and Ottonello (1997)). All the aminoacyl-tRNA can belong to any one of the following three groups. (1) Non-cognate tRNAs which have anticodon that is not compatible with the A-site codon, and these tRNAs rapidly leave the ribosome. (2) Near-cognante tRNAs whose base-pairing properties enable them to undergo part of the reactions and occupy the A-site a bit longer before dissociation. (3) Cognate tRNAs which have anticodon that forms Watson-Crick basepairs or wobble-base pairs (Sesma and Von der Haar (2014)).

In the process of elongation, when the first ribosome moves away from the start codon about 40 nucleotides, the second ribosome attaches to the start codon and begin another protein translation.

(3) Termination: When the ribosome slides along the mRNA and its 'A' site meets triplet UAA, UAG or UGA (stop codon), the release factor attaches to the stop codon. The polypeptide chain and tRNA release from the ribosome. The ribosome is disassembled from the mRNA and becomes free for the next round of initiation.

Proteins begin to fold within the polypeptide once they are located within the exist tunnel of the ribosome, which is called co-translational protein folding (Thommen, Holtkamp and Rodnina (2017)). Protein folding structure include: Secondary structure $\alpha$ helix or $\beta$ pleated sheet, a repeating pattern caused by hydrogen bondings between peptide backbones; Tertiary structure formed by side chain hydrophobic interactions; Quaternary structure is grouped by multiple polypeptide chains.

In bacteria, translation occurs in the cell's cytoplasm. In eukaryotes, translation occurs in the cytosol or on the endoplasmic reticulum (ER) (an organelle in eukaryotic cells). In many instances, the entire ribosome/mRNA complex binds to the outer membrane of the rough ER; the newly created polypeptide is stored inside the ER for later vesicle transport and secretion.

(4) Post-translation Modification: The protein precursor translated from the mRNA is usually biologically inactive. Precursor modifications generally include: Remove methionine (in eukaryotic); Cleavage of unnecessary peptides; Disulfide bonds formation which is necessary for most functional proteins and ect.

Only properly folded and assembled proteins are transported from the rough ER to the Golgi complex (an organelle in eukaryotic cells) and ultimately to the cell surface or other final destination. Unfolded and misfolded proteins are transported back into the cytosol and degraded by proteasomes.

## 2.2 Understanding Codon Usage Bias

After elaborating protein biosynthesis, we review terms relevant to codon usage bias: genetic code, codon degeneracy, synonymous codon, codon usage bias.

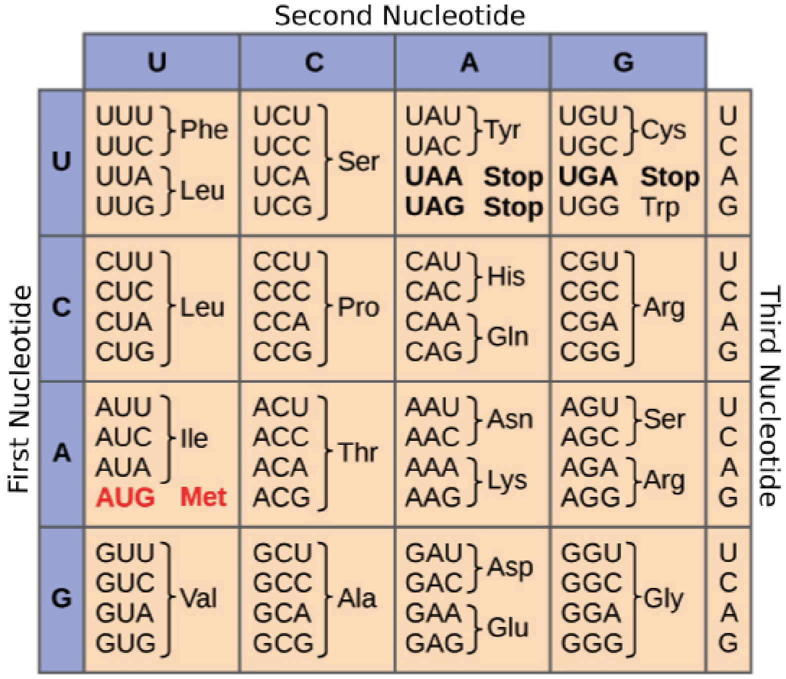
Each mRNA sequence is read from its 5' to 3' direction, from an initial nucleotide triplet to a stop nucleotide triplet. Each triplet between the position of initiation and termination is mapped to an amino acid. Such triplet nature was revealed by Nirenberg who received the Nobel prize for this discovery in 1968 (Nirenberg et al. (1965)). The mapping relationships between those triplets and amino acids can be displayed as DNA/RNA tables (Shu (2017)).

There are 20 naturally occurring amino acids (see Table 1), whose existence was investigated as early as 1952 in the Miller-Urey experiment (Johnson et al. (2008)).

Table 1: 20 Standard Amino Acids and Standard Abbreviations

| Amino Acid | 3-Letter-Abbreviation | 1-Letter-Abbreviation |
|---|---|---|
| Arginine | Arg | R |
| Serine | Ser | S |
| Leucine | Leu | L |
| Glycine | Gly | G |
| Valine | Val | V |
| Alanine | Ala | A |
| Threonine | Thr | T |
| Proline | Pro | P |
| Isoleucine | Ile | I |
| Aspartic acid | Asp | D |
| Lysine | Lys | K |
| Asparagine | Asn | N |
| Cysteine | Cys | C |
| Tyrosine | Tyr | Y |
| Phenylalanine | Phe | F |
| Glutamine | Gln | Q |
| Histidine | His | H |
| Glutamic acid | Glu | E |
| Methionine | Met | M |
| Tryptophan | Trp | W |

The mappings between codons and the 20 amino acids in the majority organisms is called standard genetic code, displayed in Figure 4. There exist genetic code variants in some organisms, for example UGA encodes tryptophan in Mycoplasma species, and CUG encodes serine rather than leucine in yeasts (Fitzpatrick et al. (2006), Santos and Tuite (1995)). In addition special code is used to encode two important proteinogenic amino acids pyrrolysine and selenocysteinein, and expanded genetic code is used for unnatural amino acids in synthetic biology (Wang, Parrish and Wang (2009)). In this work we aim to explore the universal principles across species therefore we focus on the standard genetic code.



| First Nucleotide | Second Nucleotide: U | C | A | G | Third Nucleotide |
|---|---|---|---|---|---|
| U | UUU Phe | UCU Ser | UAU Tyr | UGU Cys | U |
|  | UUC | UCC | UAC | UGC | C |
|  | UUA Leu | UCA | **UAA Stop** | **UGA Stop** | A |
|  | UUG | UCG | **UAG Stop** | UGG Trp | G |
| C | CUU Leu | CCU Pro | CAU His | CGU Arg | U |
|  | CUC | CCC | CAC | CGC | C |
|  | CUA | CCA | CAA Gln | CGA | A |
|  | CUG | CCG | CAG | CGG | G |
| A | AUU Ile | ACU Thr | AAU Asn | AGU Ser | U |
|  | AUC | ACC | AAC | AGC | C |
|  | AUA | ACA | AAA Lys | AGA Arg | A |
|  | **AUG Met** | ACG | AAG | AGG | G |
| G | GUU Val | GCU Ala | GAU Asp | GGU Gly | U |
|  | GUC | GCC | GAC | GGC | C |
|  | GUA | GCA | GAA Glu | GGA | A |
|  | GUG | GCG | GAG | GGG | G |

Figure 4: The standard genetic code. Amino acids can be grouped into families depending on how many codons encode them: One codon (Met, Trp), two codons (Phe, Tyr, His, Gln, Asn, Lys, Asp, Glu, Cys), three codons (Ile), four codons (Val, Ala, Pro, Thr, Gly) and six codons (Leu, Ser, Arg).

Along mRNA the triplet grouped by three adjacent nucleotides decoding a particular amino acid is called a codon (which has been introduced in section 1.1). Four bases constitutes mRNA (A, U, C, G), and hence there exist 64 different codons, including the start codon (AUG, encoding amino acid methionine), and the stop codon (UAA, UAG, UGA).

The multiplicity of three-base pair combinations in a codon that specifies the same amino acid is called codon degeneracy. Codon degeneracy results in the redundancy of the genetic code.

Codons encoding the same amino acid are called synonymous codons. All amino acids, except methionine and tryptophan, can be encoded by two to six synonymous codons.

Based on this one would expect that all synonymous codons are equally used. However biologically synonymous codons are not equally used within an organism and different organisms have their own preference to certain synonymous codons, as discovered five decades ago (Clarke (1970)).This phenomenon is called codon usage bias (CUB for short). For example in human genome, there are two synonymous codons CAA and CAG to encode amino acid glutamine (Glu), whose frequencies are 0.27 for CAA and 0.73 for CAG individually. When it comes to bacteria *Acidobacteria* the ratio is 0.12 for CAA and 0.88 for CAG, compared to fungi *Candida auris* the ratio between CAA and CAG is 0.46 to 0.53 (information from codon usage database [1], whose resource data is from NCBI-GeneBank). Various methods are proposed to measure such bias and also possible factors responsible for such bias are widely investigated.

## 2.3 Current Measures for Codon Usage Bias

In the previous sections we introduced the biological background pertinent to codon usage bias. In this section we summarize achievements from current researches in two aspects: measures of codon usage bias, and hypothesis for the codon usage bias origins.

There is a rich diversity of CUB measures serving for different purposes adopting different methodologies. Some of the measures aim to quantify CUB of a single gene, while others are interested in CUB of a whole genome; Some of the measures take into consideration of environmental situations the genome lies in such as the tRNA pool, while some focus on intrinsic attributes of the genome such as genomic spatial shape. The investigated perspectives range from biology, medicine, statistics, phylogenetics to physics. Therefore which CUB measure to choose depends on both the purpose of the study and the feasibility of the methodology.

For a systematical understanding, we classify these measures into two categories not according to their historical similarities but their application limitations: (1) Measures requiring external biological information besides gene sequences; (2) Measures based on the intrinsic sequence composition.

Quantification parameters of CUB are at different levels such as codon level, amino acids level, sequence level and genome level. If the approaches first compute the contribution to codon usage bias of each codon type or each amino acid type, there exists necessity to discuss how the basic quantified unit can be combined properly to assess CUB at the level of the whole gene or genome. This aspect has not been clearly summarised in previous references and hence we contribute such

[1] https://www.kazusa.or.jp/codon/

descriptions in our CUB measures summarisation.

### 2.3.1 Measures Requiring External Biological Information

In this category we introduce widely used methods which require external references depending on biological assumptions. Generally speaking these measures mainly differ in the standards of choosing reference sets.

Codon adaption index (CAI) shows codon usage bias of a gene, which is a long used and widely understood measurement first proposed in 1987 (Sharp and Li (1987)). It uses a reference set of highly expressed genes which are assumed to be under strong translation selection. For CAI calculation one first needs to calculate the relative adaptiveness ($w_i$) for each synonymous codon according to $w_i = f_i/f_{max}$, where $f_i$ is the frequency of codon i in the observed gene and $f_{max}$ is the highest frequency among synonymous codon family of codon i in the reference set. After having all the $w_i$, the combination of $w_i$ for quantification of a whole gene is $CAI = (\prod_{i=1}^{L} w_i)^{(\frac{1}{L})}$ (where L is the length of the gene sequence) the geometric mean of all the $w_i$.

Later improvements to CAI include: taking into consideration of the irregular cases which cause errors such as amino acid encoded by only one codon (Xia (2007)); Relative codon adaption index (rCAI) is proposed to better discriminate between highly biased and unbiased regions (Lee et al. (2010)). rCAI adjusts relative adaptiveness of each codon ( $w_c^{rel}$) by normalising two reading frames (+1 and +2 reading frames) as $w_c^{rel} = \frac{w_c^0}{\sqrt{w_c^{+1}}\sqrt{w_c^{+2}}}$, where $w_c = f_c/f_{max}$, and $f_{max}$ is the value derived from different reading frames.

Frequency of optimal codons (Fop) was first proposed in 1981 (Ikemura (1981)). It reflects the codon usage bias of a sequence of interest. It is the ratio between the optimal codon occurrence to the total number of codons under investigation: Fop= $\frac{O_{opt}}{O_{tot}}$, where $O_{opt} = \sum_{c \in C_{opt}} O_c$ ($O_c$ represents different codon families). The set of the optimal codons ($C_{opt}$) can be defined according to different standards such as nucleotide chemistry or tRNA availability (Ikemura (1985)).

Codon bias index (CBI) (Bennetzen and Hall (1982)) reflects codon usage bias of a sequence and is similar to Fop. But CBI defines optimal codons as codons cognate to the major tRNA species. It is calculated as $CBI = \frac{O_{\texttt{opt}} - e_{\texttt{rand}}}{O_{\texttt{tot}} - e_{\texttt{rand}}}$, $e_{rand} = \sum_{a \in A} O_a \frac{n_a^{\texttt{opt}}}{k_a}$, where $O_{\texttt{opt}}$ is the count of optimal codons, $O_{\texttt{tot}}$ is the total number of codons, $O_a$ is the occurrence of amino acid $a$, $n_a^{\texttt{opt}}$ is the optimal codon occurrence for amino acid $a$, $k_a$ is the codon redundancy. Introducing $e_{\texttt{rand}}$ for normalisation results in CBI values ranging between -1 and 1 and convenient for comparisons.

tAI (Reis, Savva and Wernisch (2004)) works by specially taking into account intracellular abundance of tRNA molecules and codon-anticodon reaction and hence it reflects codon usage bias of a gene taking into account tRNA influences. The absolute adaptiveness value $W_i$ for the $i$-th codon is defined as $W_i = \sum_{j=1}^{n_i}(1-s_{ij})CN_{ij}$, where $n_i$ is the count of cognate tRNA types for the ith codon, $CN_{ij}$ is the copy number of the jth cognate tRNA type matching the ith codon, and $s_{ij}$ is a weight representing selective constraint on the codon-anticodon coupling. Normalising $W_i$ to $w_i$: $w_i = W_i/W_{max}$. By combining all $w_i$, we obtain CUB for a gene: $tAI = (\prod_{i=1}^{l_g} w_i)^{\frac{1}{l_g}}$, where i is the $i$-th position in the gene and $l_g$ is the gene length. Affinity differences among nucleotide base pair result from physiochemical properties of nucleotides. $s_{ij}$ represents the affinity between the $i$-th codon and the $j$-th anticodon considering Crick's wobble rules, and its value is derived from biological experiments (Reis, Savva and Wernisch (2004), Watanabe and Osawa (1995)). According to tAI value we can assess the adaption of a gene to its genomic tRNA pool.

There are more tRNA interaction focused measures such as P1 index and P2 index (Gouy and Gautier (1982)). P1 index takes into account codon-anticodon interactions at the ribosome A-site: $P1 = \sum_{c \in C} \frac{f_c}{p_c}$, where $1/p_c$ is the mean number of tRNA-mRNA interactions at the A-site, $f_c$ is the codon frequency in the sequence, $C$ is the codon type space of the sequence. While P2 index accounts for codon-anticodon interactions along the whole sequence: $P2 = \frac{O_{\texttt{wwc}}+O_{\texttt{ssu}}}{O_{\texttt{wwy}}+O_{\texttt{ssy}}}$, where $w$ represents A or T, $s$ represents G or C. The underlying validation of P2 index is that pyrimiding-ending codons have intermediate strength compared to purine-ending codons, and hence the third position of nucleotide has stronger bias for G or C if the first two nucleotides have weak binding with A or T.

Codon usage bias (B) (Karlin, Mrázek and Campbell (1998)) is a parameter to assess codon usage bias of a gene. It compares the test gene set to a reference set where the reference can be composed of a gene class, an entire genome, or a single gene. Its value is based on the distance between codon frequency vectors, and is adjusted by a weight representing amino acid frequency: $B = \sum_{a \in A} F_a d(f_a, f_a^{\texttt{ref}})$, where $F_a$ is the frequency of the amino acid $a$ in the test set, $f_a$ and $f_a^{\mathtt{ref}}$ are codon frequency vectors for such amino acid $a$ in the test and reference set respectively, d is the norm distance between the two codon frequency vectors. Later analogous measures adopt different forms of distance between codon frequency vectors, such as using square distance instead of normal distance (Gladitz et al. (2005)) and adopt combined reference set such as linear combination of genes (Karlin and Mrázek (2000)).

Codon-enrichment correlation (CEC) (Ghaemmaghami et al. (2003)) indicates linear correlation between the investigated sequence and the reference set. It was originally developed for distinguishing bona fide coding regions, whereby the reference set contains real coding sequence with high confidence. The linear correlation coefficient between an interested open reading frame (ORF) and a reference set is calculated as $CEC = \mathtt{corr}(E^{\mathtt{orf}}, E^{\mathtt{ref}})$, where E is a vector containing elements of Ec for different codon types. Ec is calculated as $Ec = \frac{f_c}{e_c}$, $e_c = b_1 b_2 b_3$, where $f_c$ is the codon frequency in the investigated ORF, $b_i$ $(i = 1, 2, 3)$ are the nucleotide frequencies under a certain nucleotide distribution.

### 2.3.2 Measures Based on Intrinsic Sequence Composition

In this category we introduce widely accepted methods based on intrinsic sequence composition.

Effective number of codons (ENc) (Wright (1990)) is a simple but an effective quantification for codon usage bias of a sequence, which is independent of the gene length and the amino acid composition: $ENc = Nc_{Ala} + Nc_{Arg} + ... + Nc_{Val}$. ENc is a summation based on CUB of each amino acid type, and its final value ranges from 20 to 61. In the extreme bias when only one particular codon is exclusively used for each amino acid ENc value is 20. On the contrary ENc value is 61 if all the codons are adopted by the sequence. Several improvements to the original ENc have the general idea of introducing adjustment weights to minimise noises resulting from different synonymous codon family size (Marashi and Najafabadi (2004), Fuglsang (2005)).

GC content at silent sites (GC3) (Shields et al. (1988)) is a prevalent measurement of codon usage bias for a sequence. It is calculated as $GC3 = \frac{O_{\mathtt{nns}}}{O_{\mathtt{tot}}}$, where $O_{\mathtt{nns}}$ is the number of codons ending with G and C, $O_{\mathtt{tot}}$ is the total codon count.

Similar to P2 index in the first category this method also accepts that G-C pair has stronger binding than A-T pair, and hence G-C pair is more influential in the codon usage.

Improved CAI accepts that reference set can be the expression system (such as the whole genome the investigated gene exists), therefore it does not require the complete biological information of highly expressed genes (Puigbò, Bravo and Garcia-Vallve (2008)).

Relative synonymous codon usage (RSCU) (Sharp and Li (1986)) displays the bias for a single synonymous codon type, which is the ratio between observed codon frequency and expected frequency of this codon family. It is calculated as $r_{ac} = \frac{O_{ac}}{\frac{1}{k_a}\sum_{c \in C_a} O_{ac}}$, where $O_{ac}$ is the frequency of one synonymous codon type, $k_a$ is the size of synonymous codon family for the amino acid $a$. In this way $r_{ac}$ effectively diminishes the impact caused by different sizes of codon families. RSCU is 1 which indicates no bias for such codon, while greater than 1 means more frequent usages and less than 1 means infrequent usage of such codon.

Codon preference (P) (Gribskov, Devereux and Burgess (1984)) can reflect codon usage bias of a sequence. It was originally designed to use a sliding window of length L to locate genes and detect frameshift mutations. Later window size L is assigned the length the same as investigated sequence length. It is calculated as $P = (\prod_{i=1}^{L} w_c^p(i))^{\frac{1}{L}}, w_c^P = \frac{f_c}{e_c}$, $e_c = b_1 b_2 b_3$, where L is the sequence length, i is the codon position, $f_c$ is the frequency of the $i$-th codon, $e_c$ is the multiplication of nucleotide usage probabilities of the 3 nucleotide bases of the $i$-th codon.

Relative codon usage bias (RCB) (Roymondal, Das and Sahoo (2009)) and relative codon adaption (RCA) (Fox and Erill (2010)) share the same idea with codon preference (P). They all calculate theoretical codon frequency ($e_c$) by multiplication of the three nucleotide base frequencies ($b_1 b_2 b_3$): $e_c = b_1 b_2 b_3$, which assumes nucleotide distributions as underlying influence in CUB. However RCB obtains nucleotide distributions based on the expected nucleotide frequencies while RCA obtains nucleotide distributions based on randomly generated sequences which have the same GC percentage as the original sequence.

Codon-preference bias (CPB) (McLachlan, Staden and Boswell (1984)) reflects codon usage bias of a sequence and is one member of measures adopting deviations from theoretical distributions. CPB measures the degree of deviation of observed codon usage from the theoretical mean. It is calculated as $CPB = \frac{U - \bar{U}}{\sigma_U}$,

$U = -\log M(o)$, $M(o) = \frac{O_{tot}!}{\prod_{c\in C}(O_c!)} \prod_{c\in C} f_c^{O_c}$, $f_c = \frac{(\sum_{c\in SC} O_c)/O_{tot}}{sc}$, where $O_c$ is the observed codon count, $O_{tot}$ is the total codon count, $f_c$ is the expected frequqency, $SC$ is the synonymous codon family which codon $c$ belongs to, $sc$ is the size of the synonymous codon family. Distribution of $U$ ($\bar{U}$ and $\sigma_U$) are obtained by way of generating random sequences of length $O_{tot}$ and the same amino acids composition as the original sequence. To obtain distribution of $U$ is time costing especially for large values of length $O_{tot}$.

Maximum likelihood codon bias (MCB) (Urrutia and Hurst (2001)) borrows the same idea as CPB to compare the observance to expected distribution, but differ both in ways of expected value calculation and comparison procedure. It is calculated as $MCB = \sum_{a\in A} \frac{B_a log O_a}{O_{tot}}$, $B_a = \sum_{c\in C_a} \frac{(O_c-e_c)^2}{e_c}$, where $O_a$ is the occurrence of amino acid $a$, $O_{tot}$ is the total count of amino acids, $O_c$ is the observed codon count and $e_c$ is the expected codon count. It computes the expected value based on nucleotide frequencies and the final form of MCB is a weighted sum over all the amino acids. The weights aim to compensate the frequently used amino acids. $B_a$ is a $\chi^2$-test statistic which evaluates the observance deviations from the expected value, which can be used to make the judgement whether observance is equivalent to the case of expected codon usage with an underlying nucleotide distribution.

Intrinsic Codon Bias Index (ICDI) (Uddin (2017)) is used to assess CUB of a sequence. It ranges from 0 to 1 where 0 signify no bias and 1 for extremely high bias. It is calculated as: $ICDI = \sum_{a\in A} F_a S_a$, $S_a = \frac{1}{K_a(K_a-1)} \sum_{c\in C_a}(r_{ac}-1)^2$, where $r_{ac}$ is the synonymous codon usage frequency, $K_a$ is the degeneracy of amino acid $a$ in the sequence, $S_a$ reflects the CUB assessment at each amino acid level, and $F_a$ is an equal weight 1/18 for all the amino acid.

Weighted sum of relative entropy ($E_w$) (Suzuki, Saito and Tomita (2004)) is a sequence CUB assessment parameter originating from information theory. It measures the deviation of observances from equal codon usage cases. It is calculated as $E_w = \sum_{a\in A} F_a E_a$, $E_a = \frac{H_a}{max(H_a)}$, $H_a = -\sum_{c\in C_a} f_{ac} log_2 f_{ac}$, where $F_a$ is the relative frequency of amino acids in the sequence, $H_a$ is the entropy defined in information theory, $f_{ac}$ is the frequency of each codon. $E_w$ first computes unit assessing parameter $E_a$ and then sum $E_a$ over all the amino acids. Weights $F_a$ work as adjustments considering different amino acid frequencies.

Synonymous codon usage order (SCUO) (Wan et al. (2004)) is also an information theory based CUB measurement. It is similar to $E_w$, but differs in how the entropy of each amino acid is calculated shown as $E_a = \frac{max(H_a - H_a)}{max(H_a)}$.

Kullback-Leibler codon information bias (KL-CIB) is another sequence CUB measurement based on conditional entropy. It evaluates departure of observance from assumption of uniform synonymous codon usage. It is calculated as $K(\mu \,|\, \nu) = \sum_{i \in I} \mu(i) \log\big(\mu(i)/\nu(i)\big)$, where i is ith codon, $\nu(i)$ is the observed codon entropy distribution, $\mu(i)$ is the assumed codon entropy distribution which is derived from sampled sequences (Comeron and Aguadé (1998)).

Codon pair score (CPS) (Coleman et al. (2008)) shows codon usage bias considering context in the neighbourhood, namely *codon context bias*. It is calculated as $CPS = \frac{F(AB)}{\frac{F(A)F(B)}{F(X)F(Y)} F(XY)}$, where AB is the codon pair coding the amino acid pair XY, F is the counts for the corresponding element in the parenthesis. Arithmetic mean of CPS is used to reflect codon context bias in a quiry gene .

### 2.3.3 Comparisons among Different Measures

Without exception each measurement first obtains codon counts under certain standards. After codon counts are prepared, all the methods need to resolve two common issues 'normalisation' and 'combination'.

'Normalisation' makes the CUB assessment parameters comparable, and 'combination' transforms unit CUB assessment parameters at codon level or amino acid level to the level of the whole sequence or genome. For 'normalisation', one way is choosing reference sets which require external biological knowledge (Sharp and Li (1987), Reis, Savva and Wernisch (2004), Karlin, Mrázek and Campbell (1998), Bennetzen and Hall (1982)); the other way is comparing with theoretical distributions, some based on given nucleotide distributions (Gribskov, Devereux and Burgess (1984)), some based on the statistics derived from randomly generated sequences (Roymondal, Das and Sahoo (2009), McLachlan, Staden and Boswell (1984), Urrutia and Hurst (2001)). For 'combination', some adopt weighted summation (Uddin (2017), Suzuki, Saito and Tomita (2004)), some adopt geometric mean (Sharp and Li (1987), Reis, Savva and Wernisch (2004)), some adopt distance between vectors (Karlin, Mrázek and Campbell (1998), Ghaemmaghami et al. (2003)), and some adopt differences between distributions (McLachlan, Staden and Boswell (1984), Comeron and Aguadé (1998)). These procedures aim to diminish impact caused by sequence lengths, amino acid compositions, and synonymous codon family sizes.

Herein we compare previously introduced measures according to their different specifics, utilities and their procedures of 'normalisation' and 'combination'.

- Reference from biological information: Codon adaption index (CAI) uses highly expressed genes as the reference set. Codons adopted by highly expressed genes are considered as optimal codons, because highly expressed genes are assumed under stronger selection for desired translation efficiency. The reference set selection standard renders CAI the capability to evaluate CUB from the perspective of translation efficiency. Frequency of optimal codons (Fop) and Codon bias index (CBI) define reference set and optimal codons according to certain principles such as nucleotide chemistry, tRNA availability. By way of choosing particular reference sets, Fop and CBI are appropriate to analyse relationships between CUB and interested factors. tAI, P1 index, and P2 index are competent in evaluation between CUB and tRNA related factors because their reference set selections refer to tRNA copies, cognate tRNAs, mRNA copies, codon-anticodon affinity, and tRNA interaction with ribosome 'A' position, which are important elements in translation initiation and translation elongation. These measures are capable of analysing concrete biological factors by choosing pertinent reference sets, however due to incompleteness of available biological information, the reference set selection has inevitable restrictions when it comes to analysing the large number of emerging genomes from poorly studied organisms. To enhance their usage flexibility such measures add ways of selecting reference sets, for example codon adaption index (CAI) not only accept the highly expressed gene as a reference set but also accept the whole genome the investigated gene exists as the reference set.

- Reference from theoretical calculations. There are generally two classes of theoretical distributions: one is the expected distribution based on randomly generated sequences, and the other is based on intrinsic nucleotide distribution of the organism. Codon-preference bias measure (CPB), information theory based weighted sum of relative entropy (Ew) and synonymous codon usage order (SCUO) measure codon usage pattern deviation from expected distribution assuming codon usage are completely random. However if sequence length is too long it is difficult to get the accurate codon usage distribution. Codon preference (P) and maximum-likelihood codon bias (MCB) take another hypothetic distribution calculated based on the given nucleotide frequencies, which supposes CUB has an underlying dependance on nucleotide composition. There are mixtures of these two classes such as relative codon usage bias (RCB), which uses multiplication of the three nucleotide frequencies of a codon as the expected distribution, however the nucleotide frequencies are the expected values calculated basing on quantities of randomly generated sequences. Any assumptions a CUB measurement take could enhance the capability of evaluating a particular factor (such as nucleotide composition in this case), but meanwhile undermine the capability to spot other potential correlated factors. These measures without references from concrete biological information are more flexible to use and contain more information about potential influential factors in CUB, however convincing mechanisms are required to explain and validate results from such measures. They often make comparisons with measures of clear biological indications such as codon adaption index (CAI), in order to demonstrate their strengths in analysing biological events.

- Sequence length: Sequence length is an important issue which should be tackled properly in all the measures. Short sequences prone to have large variance due to stochastic sampling effects which appear in codon adaption index (CAI), frequency of optimal codons (Fop), codon bias index (CBI) and effective number of codons (ENc) (Behura and Severson (2013)). Weighted sum of relative entropy (Ew), synonymous codon usage order (SCUO), and codon-preference bias (CPB) all have a problem to achieve the accurate distributions to calculate the expected value when the sequence lengths are too long.

- Amino acid composition: Intrinsic codon bias index (ICDI) assesses CUB of the whole sequence by summing over all the amino acids with an equal weight 1/18, but it has taken into account amino acid degeneracy when assessing CUB for each amino acid type. Codon usage bias (B), weighted sum of relative entropy (Ew), and synonymous codon usage order (SCUO) apply amino acid frequencies in the query set as the adjustment against different amino acid compositions for CUB assessment of the whole sequence.

- Synonymous codon family size: Effective number of codons (ENc) is popular because of its simple calculation and easy interpretation. Unlike other methods it avoids complex normalisation, however it doesn't take into account different sizes of synonymous codon families. Relative synonymous codon usage (RSCU) adopts a weight to diminish the affect from the synonymous codon family size, whereas it has no proposal to combine the unit CUB assessment at the codon level for the whole sequence CUB quantification.

- Combination method: Codon adaption index (CAI) and codon preference (P) use geometric mean to combine unit CUB assessment parameter, and sequence length L is involved in the root to diminish length impact. Codon-enrichment correlation (CEC) and codon usage bias (B) choose the distance between codon frequency vectors as the final assessment parameter for a whole sequence, which is a good way to reduce dimensions and avoid improper linear combination of basic CUB assessment parameters. Effective codon numbers (ENc), maximum-likelihood codon bias (MCB), and weighted sum of relative entropy (Ew) resort to weighted sum as the combination method, where weights are used to counteract noises caused by amino acid composition.

From the above discussion of current popular CUB measures, we could see the weakness of each individual measure from the perspective of the convenience and computational cost of its application, and information loss during its calculation procedure. In this work we aim to propose a new computationally efficient measure which only relies on the intrinsic feature of the genome without external references, and furthermore maintains as complete CUB information as possible at levels of genes, amino acids and the whole genome.

## 2.4 Hypotheses for the Origins of Codon Usage Bias

In the last section, we introduced the measures for codon usage bias, which provide the tool to discover correlation between codon usage patterns and their potential causative factors. In this section, we discuss hypotheses about the origins of codon usage bias.

Underlying mechanisms responsible for codon usage bias have been widely studied but no unanimous agreement has been reached (Duret (2002)).

Codon evolution starts from nucleotide mutations, such mutations could be totally random or have certain directions. After the new mutated variants arise, the destination of the mutants are determined by two forces, which are *'genetic drift'* or *'natural selection'*. *'Genetic drift'* is a random, directionless process, and it causes the mutants to a stochastic loss or fix in a population during reproductions between generations (Duret (2008)) (When the frequency of a new mutant in the population reaches 100%, the mutant is fixed, and 0% means the mutant is lost). Such stochastic fix or loss roots from the way of offspring reproductions in a finite population. *'Natural selection'* only act on mutants which can contribute to fitness. Fitness is a relative rate of proliferation or reproduction, if the fitness of a mutant is greater than the average of the population, it will tend to increase in frequency, otherwise if less than average it will tend to decrease (Stearns and Hoekstra (2000)).

There is a widely accepted hypotheses that on a mechanistic level, two kinds of effects impacting codon usage bias are *'natural selection'* and *'mutational bias'* (Hershberg and Petrov (2008), Behura and Severson (2013), Yang and Nielsen (2008), Zeng and Charlesworth (2009)). We classify findings about CUB origins into two categories 'natural selection' and 'mutational bias' which are elaborated in the following sections.

### 2.4.1 Natural Selection

*'Natural selection'* states that codon usage undergoes positive selection. Possible selection pressures on CUB are discussed as follows:

#### 2.4.1.1 Selection Pressure Arising from Translation

Translation efficiency is highly correlated with codon usage bias. Conversions of abundant codons to their rare synonymous counterpart in several highly expressed genes shows a reduction of both the cellular fitness and the translation efficiency, which supports the assumption in *natural selection* that codon usage of highly expressed genes was selected in evolution to maintain the efficiency of global protein translation efficiency (Frumkin et al. (2018), Jeacock, Faria and Horn (2018), Tuller et al. (2010), Nakahigashi et al. (2014)). Optimal and rare codons impact translation efficiency, however there is controversy about whether rare codons undergo selection. Some state that selection favours optimal codons over rare codons, mutational pressure and genetic drift allow the rare codons to persist (Hershberg and Petrov (2008)), while some state that rare codons can contribute to the accuracy of translation although at the expense of speed (Gingold and Pilpel (2011), Cope, Hettich and Gilchrist (2018)). Translation has a trade-off between speed and accuracy (Thompson and Karim (1982), Lovmar and Ehrenberg (2006)), which suggests that optimal speed may not be reached if this generates an unacceptable accuracy price.

Codon usage bias influences the translation initiation rate by shaping mRNA secondary structure ((Liu et al. (2017))). CUB in the 5' terminal of coding sequences can result in profound effects on gene expression, because 5' end secondary structure of mRNA considerably affect translation initiation rate by controlling ribosome binding to mRNA. Besides mRNA secondary structure, codon usage largely shapes mRNA abundance by controlling mRNA decay (Erben and Clayton (2018)). Different mRNA abundances influence translation initiation rate. The translational efficiency of an AUG, CUG, GUG, or UUG start codon is measured in the naturally leaderless mRNA (the mRNA lacking 5' untranslated regions) from bacteriophage, and it suggests that the start codon is an important determinant of ribosome binding strength to mRNA (O'Donnell and Janssen (2001)).

Codon usage bias is correlated to factors involved in translation elongation such as tRNA pools, cognate tRNA abundance, aminoacyl-tRNA synthetases and ribosomes. Highly expressed genes in the *C.elegans* genome show that translation elongation rates are faster along transcripts if codons highly adapt to the tRNA pools (Duret (2000)).The relative abundance of cognate tRNA is correlated with optimal codons and rare codons, which demonstrates as a common principle in a wide range of organisms (Behura and Severson (2011), Fluitt, Pienaar and Viljoen (2007),Roy et al. (2015)). When it comes to the ribosome, another important factor engaging in translation elongation, ribosome profiling technique is adopted to detect the translation elongation velocity at the codon-level by spotting ribosome location, and it reveals that codon usage controls ribosome pausing and traffic on mRNA (Agashe et al. (2012)).

Dynamic changes to protein structure during synthesis are related to CUB. Protein folding in vivo happens simultaneously with the translation elongation, and the protein kinetic folding is effectively manipulated at the codon level (Thommen, Holtkamp and Rodnina (2017)). Synonymous codon usage impacts the speed when polypeptides emerge from the ribosome and control protein dynamic structure to avoid unwanted interactions between chemical groups (Angov (2011)). Substitutions of rare synonymous codons to mRNA templates when keeping the similar mRNA and protein abundance levels, yields altered protein conformations between the wild type and mutant protein products, which is the effect of CUB on protein kinetic folding (Kimchi-Sarfaty et al. (2007)).

#### 2.4.1.2 Selection Pressure Arising from Transcription

mRNA is produced at the transcription stage, therefore more studies extend their interests into CUB correlated factors involved in transcription.

NTPs (reference to section 1.1 transcription) serve as resources and energy for transcription, their abundances impact transcription efficiency. The most frequently used ribonuclectide at the third codon position in mRNA are the same as the most abundant NTPs in the cellular matrix where mRNA is transcribed, which provides the evidence that transcription efficiency shape codon usage (Xia (1996)).

Exonic splicing enhancers (reference to section 1.1 post-transcription modification) are parts of exons, and are responsible for proper identifications of splice sites in a primary mRNA. Large parts of functional exonic splicing enhancers are composed of codons with 4-fold degenerate sites (if a base yields the same amino acid no matter of A,U,C,G in the third codon position, such base is called 4-fold degenerate site). Exonic splice regulation imposes strong selection at synonymous sites (Savisaar and Hurst (2018)).

Transcription factors footprinting (a technique to study interactions between nucleotide sequences and proteins) across the human exome in 81 diverse cell types, shows there are highly conserved dual-use codons which simultaneously specify both amino acids and TF recognition sites. TF-imposed constraint appears to be a major driver of codon usage bias (Stergachis et al. (2013),Goz, Zafrir and Tuller (2018)).

#### 2.4.1.3 Selection Pressure from the Environment

Viruses express their protein by combining their genetic information into hosts, therefore viruses are ideal subjects to study codon evolution in particular environments. The study of 2625 different viruses and 439 corresponding host organisms uncovered that long substrings of nucleotides in the coding regions of viruses often repeat in the corresponding hosts from all domains. The host-repeating strings in the viruses resulted from the evolutionary pressure enable viruses to effectively interact with host's intracellular factors and efficiently escape from the hosts' immune system (Goz, Zafrir and Tuller (2018)). The investigation of translation kinetics and capsid folding in hepatitis A virus (HAV) showed that codon usage in HAV is highly biased and deoptimized with respect to its host, for the reason that HAV avoids using abundant host cell codons and hence eludes competition for the corresponding tRNAs (Pintó et al. (2018)).

Not only viral genomes show codon usage adaptions to their hosts, also bacteria have their codon usage strategies. Acidophilic bacteria preferentially have low CUB, which is consistent with their slow growth rate and their capacity to live in a wide range of habitats. Their codons which encode proteins to resist extreme conditions (such as metal and oxidative stress) have particular low CUB. Such results uncovered codon adaptations to environmental conditions in an acidophilic consortium (Hart et al. (2018)).

Codon optimization in fungal parasites at the genome scale correlates with their host range. The longer proteins encoded by broad host range fungi prone to have a stronger codon optimization. By contrast to the specialist species, generalist species tend to have the virulence genes which are highly codon-optimized (A generalist species is able to thrive in a wide variety of environmental conditions, while a specialist species can only thrive in a narrow range of environmental conditions) (Badet et al. (2017)).

#### 2.4.1.4 Selection Pressure Arising from Pathways

Amino acids that share the same biosynthetic pathway tend to have the same first base in their codons, and also amino acids with similar steric, chemical and physical properties tend to have similar codons (Wong (1975)). Amino acid properties correlate to nucleotide base type and base positions within the codon (Taylor and Coates (1989)). These scenarios convey that codon co-evolved with amino acid metabolic pathway.

In bacteria kingdom, at least two groups of functionally distinct genes are characterised by different levels of conservation and CUB. The first group is mainly related to cellular information processing, and it retains a limited synonymous codon usage repertoire under the purifying conservative selection. The other group is mainly related to metabolism and has less conserved codon usage (Dilucca, Cimini and Giansanti (2018)).

In the mammalian peripheral neurons, genes involved in the DNA damage repair pathway are codon-biased, and their misregulation is correlated with elevated levels of DNA damage (Goffena et al. (2018)).

#### 2.4.1.5 Selection Pressure Arising from Codon Spatial Location in the Genome

Codon location within the genome influences codon usage bias. An intra-genic variation of codon usage indicates codon usage bias is position-specific (Behura and Severson (2013)), which is to say the magnitude and direction of codon bias can vary along the gene. Nearly symmetric M-shaped spatial pattern of CUB exists among the genes of the fruit fly, with relative less CUB in the middle and the ends of the gene (Qin et al. (2004)). Slow codons are chosen at the start of the coding regions aiming to slowly load ribosomes and to avoid congestion (Tuller et al. (2010)). Codons at the intron-exon junctions have different selection pattern of codon usage compare to other exon regions (Parmley, Chamary and Hurst (2005)).

Codon context patterns and codon usage bias in a genome-wide manner analysis among insect species showed that specific codons are frequently used near the 3' prime and 5' prime in the context of the start and the stop codon (Behura and Severson (2012)). Similarly genomic and transcriptomic data of a red yeast species reveals that independent of the culture conditions, the highly expressed genes show a strong bias in the 3' context (Baeza et al. (2015)). Comparison between a key enzyme-coding gene in a bacterial wild-type and its synonymous variants, indicates that an individual gene can either select for or against particular synonymous codons depending on their local context (Agashe et al. (2012)). Adjacent codon usage patterns demonstrate that two consecutive rare codons are generally avoided, for the reason that it could increase the probability of ribosome drop-off (Cruz-Vera et al. (2004)).

Factors belonging to *Natural selection* responsible for CUB are not isolated to each other, for example codon usage bias is related to mRNA secondary structure, mRNA secondary structure influences translation initiation rate, translation initiation contributes to protein synthesis speed (protein abundances), and functional protein expression is vital for the organism fitness (survival and replication). Bacteria fitness depending on codon usage bias is frequently confirmed in many microbiological experiments (Hauber, Grogan and DeBry (2016),Yannai, Katz and Hershberg (2018)).

## 2.4.2 Mutational Bias

*'Mutational bias'* favours certain types of mutations, which can be caused by the chemical properties of the nucleotide bases (Knight, Freeland and Landweber (2001)), non-uniform DNA repair (Kaufmann and Paules (1996)), non-random replication errors (Lobry (1996)) etc.. Types of mutational bias include GC bias; transition-transversion bias; strand-specific bias; insertion-deletion bias, among which GC bias is the most widely discussed as a driver for CUB. All kinds of mutational variants are assumed to arise randomly and provide the raw materials for evolution (Whitehead and Crawford (2006)). Further more mutational bias could potentially cause codon bias.

### 2.4.2.1 GC Bias

GC (guanine-plus-cytosine) bias leads to directional mutations. GC pairs has a higher thermostability compared to AU/AT pairs, because the GC pair has three hydrogen bonds while the AU/AT pair has two. Besides GC pairs have more favorable stacking energies (Yakovchuk, Protozanova and Frank-Kamenetskii (2006)).

DNAs/RNAs appear to be biased towards the preferential fixation of AT/AU to GC mutations, and hence GC content is considered as a mutational bias force driving codon usage (Birdsell (2002)). Supportive viewpoint states that a guanine- and cytosine-rich genome is preferred from an evolutionary standpoint (Nabiyouni, Prakash and Fedorov (2013)). GC bias puts a directional pressure on the genome to evolve towards a preferred GC content, and these directional changes happen more in neutral parts of the genome than in functionally significant parts (Sueoka (1988)).

GC bias shapes the codon usage at the global level in the genome. AT to GC mutations sculpt nucleotide compositions, and the GC content correlates with non-coding, exon, intron, tRNA and rRNA sequences, all parts of the genome (Muto and Osawa (1987)). GC1, GC2, GC3, and the whole GC content (1,2,3 represent the nucleotide position within the codon) are often taken into consideration separately in studies (Du et al. (2018), Mondal et al. (2016)). Such global mutation force results in nucleotide-composition-shaped codon usage bias, which is more likely determined by global genome-wide processes rather than selective forces acting specially on gene sequences (Knight, Freeland and Landweber (2001),Guo, Bao and Fan (2007), Agashe et al. (2012)).

An concrete investigation into genes involved in the central nervous system (CNS) adopted effective number of codons (ENc) and relative synonymous codon usage (RSCU) to measure CUB. The measures reveal that the most frequently occurring codons had G or C at the third position, and GC-rich genes are affected by mutation pressure (Uddin and Chakraborty (2018)). In human the expression level of oncogenes (genes of the potential to induce cancer) is determined by codon usage bias. Highly expressed oncogenes had rich GC contents with a strong codon usage bias (Mazumder, Chakraborty and Paul (2014)).

#### 2.4.2.2 Transition-transversion Bias

Nucleotide transition means purine to purine or pyrimidine to pyrimidine mutations; nucleotide transversion means purine to pyrimidine mutations or vice versa.

It is generally assumed that there is a universal bias in favour of transitions over transversions, possibly as a result of the underlying chemistry of mutations and conservative mutational effects on proteins (Stoltzfus and Norris (2015)). Methylation effect (a process of adding methyl groups to the nucleic acid molecule) is considered as the significant factor contributing to differences between transition and transversion rates. Part of the higher transition rate in vertebrates can be attributed to the effect of methylation (Keller, Bensasson and Nichols (2007)).

#### 2.4.2.3 Strand-specific Bias

Vertebrate mitochondria have circular DNAs. The circular DNA consists of two strands which have different masses because of different proportions of heavier nucleotides. Codons ending in T and G are preferentially used for heavy strand-encoded genes, and tRNAs encoded by heavy strands contain more G-U base pairs in their possible secondary structures. Accumulation of G and T on one strand, as well as A and C on the other is considered being driven by the strand-specific bias (Asakawa et al. (1991)).

#### 2.4.2.4 Insertion-deletion Bias

Deletions of nucleotides occur more frequently than insertions, which can be explained by the thermodynamics of DNA replication slippage. An insertion requires the melting and replication of a segment of previously duplicated bases, by contrast deletions only involve a skipping of unreplicated bases (Petrov (2002)).

Patterns of deletion and insertion inferred from bacterial pseudogenes demonstrate a pervasive bias towards deletions not insertions. Further more the size of deletions is biased towards the multiples of 3 nucleotides ($'3n'$). This $'3n'$ deletion pattern is explained by the alternative end-joining repair, which is a recombination-independent double strand break DNA repair mechanism (Danneels, Pinto-Carbó and Carlier (2018)).

## 2.5 Models to Investigate the Origin of Codon Usage Bias

In the previous section, we described main hypotheses about CUB origins. In this section we introduce models which are used to investigate CUB origins from the perspective of simulating dynamic codon evolution through time under assumed conditions. The current popular methods normally postulate a time frame in which the codon usage attribute changes. Various algorithms adopted to express such change include: constitution of the transition matrix which contains parameters representing all kinds of CUB driving forces; or exploration for a function whose input domain contains codon usage frequencies and possible CUB drivers. These methods have advantages to investigate particular CUB driving forces in interest however it is difficult even impossible to enumerate all the possible influential factors in the model. Throughout the thesis I developed an approach based on statistical mechanics, which summarizes the effect of large number of forces which do not need to be defined in detail. This therefore addresses the shortcomings in the existing approaches.

### 2.5.1 Model Based on Dynamic Codon Frequencies in Codon Sequences

If we consider dynamic codon frequencies at a time as the evolving events, a model can construct Markov Chains to investigate *mutational bias* and *natural selection*.

Markov chain is a mathematical system where the probability of events depend only on current states. The equilibrium status at a time of the Markov Chain depends on a *codon substitution probability matrix* ($P$, of size 61*61) (stop codon excluded).

The *codon substitution probability matrix* $P$ is changes with time and derived from *codon instantaneous substitution rate matrix* ($Q$ of size 61*61). At the start point $P(0)$ is the identity matrix the same size as $Q$, then at time t: $P(t) = Q^t$ for the discrete time, and $P(t) = e^{Qt}$ for the consecutive time ('t' represents the evolution duration or iteration).

*Codon equilibrium frequency matrix* ($\Pi$) can be calculated according to the *original codon frequency matrix* $\Pi_0$ and the *codon substitution probability matrix* at time $t$ (P(t)): $\Pi = \Pi_0 P(t)$, $\Pi_0$ is the original codon frequency distribution.

The key procedure of deriving *codon instantaneous substitution rate matrix* (Q) adopts some typical parameters as follows:

- Multiplication of involved nucleotide base mutation rates. To calculate the synonymous codon mutation rates, most models only consider one base mutation (Yang and Nielsen (2008), Pouyet et al. (2016)), while some accept di-base mutations (Cannarozzi and Schneider (2012)). Among the four different nucleotides 'A, T, C, G' , the mutual pair mutation rates form a 'nucleotide mutation rate matrix' ($N$ of size 4*4). In a computational model the pair mutation rates can be equal to each other under a total random mutation assumption. Based on the matrix N, synonymous codon substitution rates solely depending on directional nucleotide mutations are obtained.

- Parameters rooted in biological considerations and principles such as: $'\omega'$ , the ratio between the rate of synonymous mutations (dS) and the non-synonymous mutations (dN); $'k'$, the ratio between the rate of nucleotide transitions and the rate of nucleotide transversions (Nucleotide transition means purine to purine or pyrimidine to pyrimidine mutations; nucleotide transversion means purine to pyrimidine mutations or vice versa); $'d4'$: the rate at the 4-fold degenerate sites (if a base yields the same amino acid no matter of A,U,C,G in the third codon position, such base is called 4-fold degenerate site) (Zoller and Schneider (2012)).

- Selected eigenvectors of $Q$ matrix in previous principal component analysis (PCA) (Jolliffe and Cadima (2016)). Eigenvectors of $Q$ matrix constitute the principal dimensions of codon substitution probability, and they retain the most principal information conveyed by matrix $Q$.

- Parameters specially adjusted for certain purposes such as phylogenetic tree branch length (Gil et al. (2013)), translation efficiency: codon fitness decided by ribosome and tRNA abundance (Bulmer (1991)).

To be specific this model could serve for the two general purposes:

(1) Probe whether mutational bias or natural selection shapes the codon usage bias of an organism. For example compare a mutational bias only model and a model containing natural selection parameters. When one only considers parameters of nucleotide substitution rate to generate synonymous codon mutation rate matrix $Q1$, the corresponding equilibrium frequency matrix $\Pi1$ is obtained. If one considers not only nucleotide substitution rate to form synonymous codon mutation rate matrix $Q2$, the corresponding equilibrium frequency matrix $\Pi2$ is obtained. Apply $\chi^2$ independence test (Yang and Nielsen (2008)) to the two

multilevel variables Π1 and Π2, with the Null Hypothesis: Π1 and Π2 are totally independent, namely reputing the null hypothesis means: the synonymous codon evolution is not caused by mutational bias only. When we apply the model to the real phylogenetic frame, orthologous groups are normally selected from species with undisputed phylogenetic trees (Cannarozzi and Schneider (2012)). Nucleotide mutation rates and all the parameters involved in the model should be retrieved from the real orthologous data.

(2) Parameter estimation for CUB drivers. First a likelihood function ($\mathcal{L}$) is written algebraically or numerically in the terms of parameters (such as nucleotide mutation rate, $'k'$ the rate ratio between nucleotide transitions and transversions, or tree branch length. For convenience we name the set of all the variables as $\theta$, and hence Q is the function of $\theta$). On the condition of observing the sequence data Π in the consecutive time, the maximum likelihood function is built: $\mathcal{L}(\theta|\Pi) = P(t) = e^{Q(\theta)t}$, where t corresponds to status Π. If t is not given, but initial $\Pi_0$ is given, then $\Pi = \Pi_0 P(t) = \Pi_0 e^{Q(\theta)t}$. By introducing Lagrange Multiplier $\lambda$, the maximum likelihood function is $\mathcal{L}(\theta, t, \lambda|\Pi, \Pi_0) = e^{Qt} - \lambda(\Pi - \Pi_0 e^{Qt})$ (in discrete time $Q^t$ should replace $e^{Qt}$). Finally maximum Likelihood Theory is used to estimate these parameters given the observable sequence data (Cannarozzi and Schneider (2012)).

Because the likelihood to see the real sequence data should have the maximum value among the parameters space, thus parameters corresponding to such maximum likelihood can be calculated by rendering the first derivative of $\mathcal{L}$ to 0 or computationally searching the maximum value of $\mathcal{L}$ among the whole available parameter space of $\theta$.

### 2.5.2 Model Based on Dynamic Allele Frequencies in a Population

Compared to the model based on dynamic compositions of a codon sequence, another type of model borrows the concept of *'genetic drift'* in population genetics. Codon usage bias is investigated from the perspective of allele frequency variations through generations among a population of interest. The fitness of an allele takes into account of codon usage such as how many optimal codons are used in the allele.

Nucleotide mutations in a gene yields its variations which are called alleles. An observable physical trait of an organism (phenotype) is decided by alleles (genotype). If two alleles together decide the phenotype of the organism, such organism is called a diploid organism (like humans). The same naming principle applies to haploid (one allele), triploid (three alleles) or polyploid (many alleles). A allele stochastically fixed or lost roots in the way the organism reproductions in the population.

Two classic models to describe allele dynamics are typically built upon the Wright-Fisher model and Moran model (Lange (2003)) under particular population genetic settings.

Wright-Fisher model assumes a gene with two alleles, A or B. The assumption is: in diploid populations of N individuals, each individual can have two copies of the same allele or two different alleles. For each generation genes contained in each individual are drawn independently at random from all gene in the parent generation. After a certain iterations frequencies of A and B can be simply calculated by binomial distribution. The probability of observing m copies of allele A is calculated as : $P\left\{X(t+1)=m|X(t)=Np\right\}=\begin{pmatrix}N\\m\end{pmatrix}p^m(1-p)^{N-m}$, where $X(t)$ denotes the number allele A at the the time $t$, p is the frequency of allele A at the time $t$. Approximation (diffusion) can be made by assuming the population size N is large and any terms in the formula of higher order of $N^{-1}$ are neglected (Tian (2007)). If there is only *'genetic drift'* acting on the population, the expected time to fix an allele is: $\bar{T}_{\text{fixed}}=\frac{-4N(1-p)\ln(1-p)}{p}$; the expected time to loss an allele is: $\bar{T}_{\text{lost}}=\frac{-4Np}{1-p}\ln p$ (Hartl, Clark and Clark (1997)).

Similarly the Moran model has the same idea as Wright-Fisher model, only differs in the aspect that it assumes overlapping generations, and at each iteration, one individual is chosen to reproduce and one individual is chosen to die (Moran (1958)). Computational simulations are usually easier to perform using the Wright-Fisher model, because fewer time steps need to be calculated (Ferrer-Admetlla et al. (2016)).

The advanced models to study codon usage bias are built on those Wright-Fisher or Moran models such as the *reversible mutation model* (Crow, Kimura et al. (1970)).

The general procedures of *reversible mutation model* to investigate *mutation* or *selection* strength are explained as follows:

(1) Assume two types of variants, $A$ and $B$ can occur among a population of size $N$. Mutation is assumed to be reversible: the mutation rate from $A$ to $B$ is $k\mu$, and that in the reverse direction is $\mu$; mutation is said to be unbiased when $k = 1$. The fitnesses of the three genotypes AA, AB, BB are 1, 1-s, and 1-2s.

(2) The system's equilibrium distribution $f_x$ is obtained using diffusion theory (Kimura (1964)): $f(x) = Cx^{\theta-1}(1-x)^{k\theta-1}e^{\gamma x}$, where $x = i/(2N)$, $\theta = 4N\mu$, $\gamma = 4Ns$, C is the constant to guarantee $\int_0^1 f(x) = 1$.

(3) When allele $A$ is fixed in the population, the likelihood of randomly choosing the allele $A$ is approximated as: $\mathcal{L} = \frac{1}{1+ke^{-\gamma}}$ (Zeng and Charlesworth (2009)).

(4) To test the effects of various covariates of codon usage in a real dataset, such as whether highly expressed genes are under stronger selection on codon usage bias in an organism, a likelihood ratio test ($LRT$) is performed ($LRT$ here is a $\chi^2$ statistic used for comparing the two likelihood variables). Find fitness values of genes from the real data to form the likelihood function. The fitness value is often referred by the codon occurrences. Assign genes with high and low expression levels to have different fitnesses of $\gamma_h$ and $\gamma_l$, then generate corresponding likelihood functions $\mathcal{L}_h$ and $\mathcal{L}_l$. Finally calculate the likelihood ratio $LR = -2log(\frac{\mathcal{L}_h}{\mathcal{L}_l})$, and p-value statistics corresponding to $LR$ in $\chi^2$ distribution is adopted to judge whether the likelihood are the same, namely whether the fitness of high expression (a high optimal codon proportion) influences the allele frequencies (codon frequencies or codon usage pattern) in the population (Cannarozzi and Schneider (2012)).

### 2.5.3 Model Based on Information Channel

A rate-distortion model based on information theory investigates how the genetic code evolved. It considers process of translating the RNA/DNA into corresponding amino acids as an error-prone information channel (Tlusty (2007)).

The information channel is featured with genetic code as its transition function. Then the error-load ($H_{ED}$) is the sum over all the average chemical differences between the desired amino acid ($\alpha$) and observed output ($\beta$): $H_{ED} = \sum_{\alpha,j,j,\beta} P_\alpha E_{\alpha i} R_{ij} D_{j\beta} C_{\alpha\beta}$ (where $\alpha$ is the desired amino acid, $\beta$ is the observed amino acid, $P_\alpha$ is the probability of requiring amino acid $\alpha$, $E_{\alpha i}$ is the probability of codon i encoding amino acid $\alpha$, $R_{ij}$ is the probability of misreading j instead of i, $D_{j\beta}$ is the probability of codon j encoding amino acid $\beta$, and $C_{\alpha\beta}$ is the chemical distance between amino acid $\alpha$ and $\beta$). In this sense, error-load serves as the fitness measure for the reliability of a genetic code in the channel. Ideally the smallest error-load scheme is the optimal genetic code.

In order to find the maximum amino acids corresponding to a codon topology, the topology coloring problem theorem is applied: the codon space is portrayed as a graph with vertices as the codons. Two codons i and j are linked by an edge if they differ only in one base. Vertex colouring results demonstrate that for 64-codons topology the maximum amino acids is 25, and for the 48-codons topology this limit is 20.

The information transition model explains how the genetic code evolves once a new amino acid emerges: the genetic code tries to use minimum codons to encode maximum amino acids meanwhile guarantee codon discernibility. Investigation into how a genetic code withstands inherent noise suggests that the three conflicting evolutionary forces interplay together to shape genetic code: the needs for diverse amino acids, error-tolerance, and minimal resource cost (Tlusty (2008)).

# Chapter 3

# A Novel CUB Measure

In the literature review, we showed that codon usage along mRNA is non-random and various measures are provided to quantify CUB. All the CUB measures start from calculating codon counts under predefined notions. Normalization aims to adopt internal or external references to normalize the measure in a comparable manner among different sequences. Combination aims to combine CUB measure at per sequence and per amino acid level into a measure at the whole genomic level which contains information of all the sequences and all the amino acids. In this chapter we will propose a novel CUB measure with multilevel normalizations and combinations, which diminishes impact by sequence lengths and minimizes the loss of CUB information carried by the whole genome during dimensionality deductions.

First we introduce a novel measure $Sn$ to analyse sequence specific codon usage bias for different amino acids. We applied $Sn$ measure to 462 sequenced fungal genomes and found that the sequence specific CUB correlates to the sequence length and they two cooperate to meet the requirement of the protein production in the cell. In addition we analysed CUB in homologs where sequences have specific relationships, and we found that CUB is broadly stronger in real genes than in genes simulated to have no codon usage bias, CUB becomes stronger with increasing choices of synonymous codons to encode that amino acid and CUB patterns are related to gene functions.

The basic core concept of our work is illustrated in Figure 5, and the main symbols adopted are listed in Table 2.

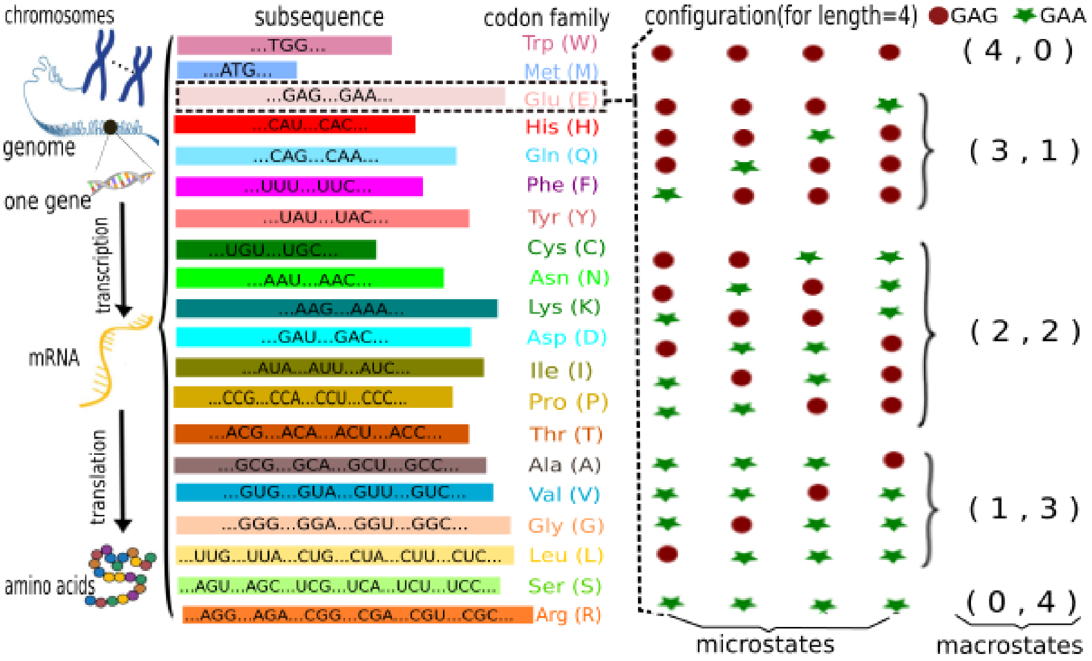


Figure 5: Basic core concepts. An mRNA sequence can be divided into different subsequences where each subsequence encodes only one amino acid type and hence is composed of one synonymous codon family. If we list the counts of different synonymous codons in the subsequence as a vector, the vector will represent synonymous codon usage pattern of such subsequence. We annotate this vector as *codon occurrence configuration*. Take a subsequence encoding Glu of length 4 as an example, such subsequence only contains synonymous codons 'GAG' and 'GAA', and all the possible codon occurrence configurations defined by 'GAG' and 'GAA' contained in such subsequence are illustrated above.

## 3.1 Mathematical Algorithm for CUB Measure

### 3.1.1 *Subsequence*

Each mRNA sequence can be divided into 20 subsequences, and each subsequence is responsible to encode one amino acid type. *Subsequence* of length $L$ encoding the amino acid $AA$ is denoted as $S^{L,AA}$.

Consider an mRNA chain as 'GAG UUU GAA GAG UUC AUA AUU GAG AUA' which can be divided into 3 subsequences:

(1) subsequence 1 'GAG GAA GAG GAG' of length 4, which encodes amino acid GLU.

(2) subsequence 2 'UUU UUC' of length 2, which encodes amino acid PHE.

| **Symbol** | **Meaning** |
|---|---|
| $C_i^{AA}$ | $i$-th codon type of amino acid AA |
| $\lvert C^{AA}\rvert \in \{1, 2, 3, 4, 6\}$ | The number of codons encoding amino acid AA |
| $n_i^{AA,g}, n_i^{AA}, n_i$ | The number of codons of type $i$ for amino acid AA in gene g. |
| $N := [n_1, n_2, ..., n_{\lvert C^{AA}\rvert}]$ | Codon occurence configuration of a subsequence |
| $L^{AA,g} := \sum_i n_i^{\mathrm{AA,g}}$ | Subsequence length encoding amino acid AA in gene g. |
| $P^{AA} := [P_1, P_2, ..., P_{\lvert C^{\mathrm{AA}}\rvert}]$ | Underlying codon usage probability for amino acid $AA$ |
| $P_N^{L,AA}$ | Multinomial distribution probability for a configuration $N$ |
| $P_{\max}^{L^{\mathrm{AA,g}}}$ | Maximum multinomial distribution probability for length $L^{\mathrm{AA,g}}$ |
| $Sn$ | Measure of codon usage bias per sequence per amino acid |
| $\overline{Sn}$ | Expected Value of all the possible $Sn$ for a length |

Table 2: Symbols adopted throughout this work

(3) subsequence 3 'AUA AUU AUA' of length 3, which encodes amino acid ILE.

### 3.1.2 *Codon Occurrence Configuration*

It is difficult to describe the exact spacial pattern of codons in a gene with the mathematical language based on original subsequence compositions, however it is much easier to depict $S^{L,AA}$ and quantify codon usage pattern based on *codon occurrence configuration* $N^{L,AA} := [n_1, n_2, ..., n_{\lvert C^{AA}\rvert}]$, where $n_i$ is the quantity of the $i$-th synonymous codon type in $S^{L,AA}$, $\lvert C^{AA}\rvert$ means how many types of synonymous codons are able to encode $AA$, $\sum_{i=1}^{\lvert C^{AA}\rvert} n_i = L$, $L$ is the subsequence length and also the number of amino acid $AA$ in a gene.

Consider an mRNA chain 'GAG UUU GAA GAG UUC AUA AUU GAG AUA' which can be divided into three subsequences:

(1) subsequence 1 'GAG GAA GAG GAG' of length 4, which encodes amino acid GLU. There are 3 'GAG' and 1 'GAA' in this subsequence, and hence the codon occurrence configuration is represented as a vector [3,1].

(2) subsequence 2 'UUU UUC' of length 2, which encodes amino acid PHE. There is 1 'UUU' and 1 'UUC' in this subsequence, and hence the codon occurrence configuration is represented as a vector [1,1].

(3) subsequence 3 'AUA AUU AUA' of length 3, which encodes amino acid ILE. There is 2 'AUA' ,1 'AUU' and 0 'AUC' and hence the codon occurrence configuration is represented as a vector [2,1,0].

For subsequence 1, codon usage pattern of 'GAG' and 'GAA' is reflected by the codon occurrence configuration $N_1^{Glu} = [3, 1]$.

There are two kinds of system states when describing subsequences: micro states and macro states as illustrated in Figure 5. The micro state refers to the original codon sequence along mRNA, and the macro state refers to the codon occurrence configuration of that original codon sequence. This work will further build a mathematical model based on macro states to describe CUB.

### 3.1.3 Multinomial Distribution Probability

Rather than depicting individual copy number of each synonymous codon type in a codon occurrence configuration, it is simpler to adopt one single value to summarise a particular macro state $N^{L,AA}$: the multinomial distribution probability $P_N^{L,AA}$ of observing $N^{L,AA}$.

For a synonymous codon family size of $m$, the number of codons encoding the same amino acid AA ($|C^{AA}|$), to construct a subsequence $S^{L,AA}$ is like performing $L$ individual trials where each trial leads to obtaining one codon from $m$ choices. If $P_i$ is the probability to choose the $i$-th synonymous codon for any trial of $L$, the probability $P_N^{L,AA}$ to observe $n_i$ copies of the $i$-th codon follows the multinomial distribution:

$$\begin{aligned} P_N^{L,AA} &= \frac{L!}{n_1!n_2!...n_m!} P_1^{n_1} P_2^{n_2} ... P_m^{n_m} \\ \sum_{i=1}^{m} n_i &= L \\ \sum_{i=1}^{m} P_i &= 1 \\ m &= |C^{AA}| \end{aligned} \tag{1}$$

where $P^{AA} = [P_1, P_2, ...P_i, ..., P_m](m = |C^{AA}|)$ is the underlying codon usage probability for each synonymous codon in its family, $|C^{AA}|$ is the size of synonymous codon family, $N^{L,AA}$=$[n_1, n_2, ..., n_m]$ is the codon occurrence configuration for the subsequence, $P_N^{L,AA}$ is the probability to observe such $N^{L,AA}$.

However $P_N^{L,AA}$ largely depends on subsequence length $L$, it can not offer a comparable quantification when the subsequence length differs largely, therefore we propose a normalisation by way of using $P_{\max}^{L,AA}$.

### 3.1.4 The Maximum Multinomial Distribution Probability

Among a group of subsequences which encode the same amino acids and have the same length, there is the maximum probability $P_{\max}^{L,AA}$ to observe the most likely codon occurrence configuration, denoted as $N_{\max}^{L,AA}$=$[n_1^{\max}, n_2^{\max}, ...n_m^{\max}]$ $(m = |C^{AA}|)$.

To obtain the maximum probability $P_{\max}^{L,AA}$ corresponding to the most likely codon occurrence configuration, rather than list all the $P_N^{L,AA}$ values for all the possible configurations, we put forward a time efficient method to get $P_{\max}^{L,AA}$ by applying Maximum Likelihood theory.

We adopt maximum log-likelihood as stated in Equation 2, which is actually to solve a optimization problem of maximising function $\ln(P_N^{L,AA})$ subject to a constrain $\sum_{i=1}^m P_i = 1$. By constructing Lagrange function $\mathcal{L}$

$$\mathcal{L}(n_1, n_2, ..., n_m, \lambda) = \ln(P_{N^{L,AA}}) + \lambda(1 - \sum_{i=1}^{m} \frac{n_i}{L}) \tag{2}$$

and then rendering the first derivative of $\mathcal{L}$ over $\lambda$ and $n_i$ to zero ($i \in [1, m]$), we obtain the most likely configuration $[n_1^{\max}, n_2^{\max}, ...n_m^{\max}]$, where $n_i^{\max} = LP_i$.

In our case codon occurrence $n_i^{\max}$ can only be integers, and hence we perform following calculation:

- Find configuration $N^0 = [n_1^0, n_2^0, ..., n_i^0, ...n_m^0]$, $(N^0 = \sum_{i=1}^m n_i^0)$, where $n_i^0 = P_i L$. If all elements in $N^0$ are integers, $N^0$ is the desired configuration for maximum $P_{\max}^{L,AA}$.

- Otherwise round each decimal $n_i^0$ down to integer $n_i^1$, and we get vector $N^1 = [n_1^1, n_2^1, ..., n_i^1, ...n_m^1]$.

- Let Lrm=$L - \sum_{i=1}^m n_i^1$, and partition Lrm into m categories $N^2 = [n_1^2, n_2^2, ..., n_i^2, ...n_m^2]$, where $Lrm = \sum_{i=1}^m n_i^2$.

- Let $n_i = n_i^1 + n_i^2$, we get vector $N = [n_1, n_2, ..., n_i, ...n_m]$. According to all the possible configuration $N$ resulted from $N^2$, we spot the maximum value as $P_{\max}^{L,AA}$.

By this efficient approach the heaviest calculation for 6 synonymous codon families is to distribute 5 into 6 categories such as [4,1,0,0,0,0], for 4 synonymous codon families is to distribute 3 to 4 categories, for 3 synonymous codon families is to distribute 2 to 3 categories, and for 2 synonymous codon families is to distribute 1 to 2 categories. To assist explanation we quote MATLAB code as follows.

```
function [Pmax,X] = EforMore(cLeng,subSleng,cfPart)
P=cfPart;
X=subSleng.*cfPart;
Xtest=mod(X,1).*10;
%decimal part, *10, if not divisable, != 0
if isempty(find(Xtest,1))
    Pmax=mnpdf(X,P);
    %calculate maximum probability
else
    Xpre=subSleng.*cfPart;
    rmv=subSleng-sum(floor(Xpre));
    %remainer value
    %  [Pmax,X] = getPmax(cLeng,cfPart,floor(Xpre),rmv);
    Pmax=getPmax(cLeng,cfPart,floor(Xpre),rmv);
    % to save time only output Pmax
end
end
```

```
% function [Pmax,Xmax] = getPmax(cLeng,cfPart,Xpre,rmv)
function Pmax = getPmax(cLeng,cfPart,Xpre,rmv)

pcount=1;
switch cLeng
    case 2
        for i=0:rmv
            j=rmv-i;
            if j>=0
                mnvect=[i,j];
%% mnvect: all the possible rmv (remain values) partitions
```

```
                X{pcount}=Xpre+mnvect;
                p(pcount)=mnpdf(X{pcount},cfPart);
                pcount=pcount+1;
            end
        end


    case 3
        for i=0:rmv
            for j=0:rmv
                k=rmv-i-j;
                if k>=0
                    mnvect=[i,j,k];
                    X{pcount}=Xpre+mnvect;
                    p(pcount)=mnpdf(X{pcount},cfPart);
                    pcount=pcount+1;
                end
            end
        end

% 4 and 6 codons use the same logic
end
[Pmax,idMAX]=max(p);
% Xmax=X{idMAX};

end
```

It remains a challenge for current CUB measures to find a non-heuristic statistical reference for normalisation (McLachlan, Staden and Boswell (1984), Suzuki, Saito and Tomita (2004), Wan et al. (2004)), but our computationally efficient method of calculating $P_{\max}^{L,AA}$ renders the ratio of $P_N^{L,AA}/P_{\max}^{L,AA}$ as an effective normalisation especially for a long codon sequence encoding the amino acid with more synonymous codon choices. We define our sequence level CUB measure $Sn$ as follows:

$$Sn = \frac{1}{L} \ln\left(P_N^{L,AA}/P_{\max}^{L,AA}\right) \tag{3}$$

where $\frac{1}{L}$ aims to normalise against the sequence length variation. Taking the logarithm aims to improve the sensitivity of the measure. We will first show that it captures relevant biological information about CUB.

### 3.1.5 Statistical Power of $Sn$

The overall approach we use is to assume as a null hypothesis that there is no overall CUB acting in genomes. In this case we would expect, from the basic considerations, that the overall distribution of particular codons obeys the multinomial distribution with $P_i = 1/m$ for $(i \in [1, m])$. Any systematic and significant deviation from this equal probability multinomial distribution signifies a CUB.

In our measure we calculate $P_N^{L,AA}$ based on observed macro states and postulated equal underlying codon usage probabilities ($P^{AA}$ with equal elements $P_i = 1/m$). If there exists CUB, $P_N^{L,AA}$ should locate outside the confidence interval of the postulated multinomial distribution, which is equivalent to saying that the difference between $P_N^{L,AA}$ and $P_{\max}^{L,AA}$ should not be explained only by chance.

To test whether such difference between $P_N^{L,AA}$ and $P_{\max}^{L,AA}$ arises by chance or not, we could apply two sample $\chi^2$ test between these two macro states to see whether they comply with the same multinomial distribution, or simply using Pearson's $\chi^2$ test to testify whether observed macro state have the postulated underlying codon usage probability.

Assuming $P_N^{L,AA}$ corresponds to the macro state $N_n^{L,AA} = [N_n^1, N_n^2, ...N_n^m]$ and hence the empirical codon usage ratio $P_n^{AA} = [\frac{N_n^1}{L}, \frac{N_n^2}{L}, \frac{N_n^1}{L}, ...\frac{N_n^m}{L}]$, the postulated underlying codon usage probability is $P^{AA} = [P_1, P_2, ...P_m]$ where $P_1 = P_2 = ...P_m = 1/m$.

Null hypothesis H0 is that the observed macro state complies with the underlying codon usage probability, namely $P_n^{AA} = P^{AA}$. The alternative hypothesis Ha is that the observed macro state does not comply with the underlying codon usage probability, namely there exists unequal entry between $P_n^{AA}$ and $P^{AA}$. Test Statistic $\chi^2 = \sum_{i=1}^{m} \frac{(N_n^i - L/m)^2}{L/m}$, degree of freedom is m-1, significance level $\alpha = 0.05$.

If the null hypothesis test is rejected, we state that $Sn$ has the statistical power to quantify CUB of the subsequence at the significance level $\alpha = 0.05$. If the null hypothesis is accepted we state that $Sn$ evaluates codon usage deviation of the observed macro state from the most probable macro state but can not exclude the reason by chance.

### 3.1.6 Theoretical $Sn$ Distribution and Expected $Sn$

To better understand $Sn$ properties we investigate expected value of $Sn$ and its distribution. The deviation of observed $Sn$ from the expected value ($\overline{Sn}$) provides a standard reference to investigate CUB of a subsequence.

For the subsequences with length $L$, suppose the $j$-th configuration $N_j$ has the multinomial distribution probability $P_{N_j}$, $P_{\max}$ is the maximum value among all the $P_{N_j}$, thus $\overline{Sn}$ of the whole population is shown in Equation 4.

$$\overline{Sn} = \frac{1}{L} \sum_{j} (P_{N_j} \ln \frac{P_{\max}}{P_{N_j}}) \tag{4}$$

where $\overline{Sn}$ is the normalised expected $Sn$ for subsequences of length $L$. $P_{N_j}$ is the probability to observe the subsequence in the state $N_j$, $P_{max}$ is the maximum probability corresponding to the most probable macro state, and thus $\ln \frac{P_{\max}}{P_{N_j}}$ reflects the difference of probabilities between the most probable macro state and the macro state of $N_j$.

Figure 6 depicts how to obtain $\overline{Sn}$ for the group of subsequences encoding an amino acid type.

Next we apply the above proposed algorithm for CUB measure to biological genome datasets, aiming to find CUB patterns based on $Sn$.

## 3.2 Generation of Datasets for CUB Measure

### 3.2.1 Processing Resource Genome Data from FASTA Files

The main data source for this project was ENSEMBL database[1]. ENSEMBL database contains genomic data and gene models from the International Sequence Database Collaboration, a collaboration that includes the National Centre for Biotechnology Information (NCBI, USA), the European Bioinformatics Institute

[1] http://fungi.ensembl.org/index.html

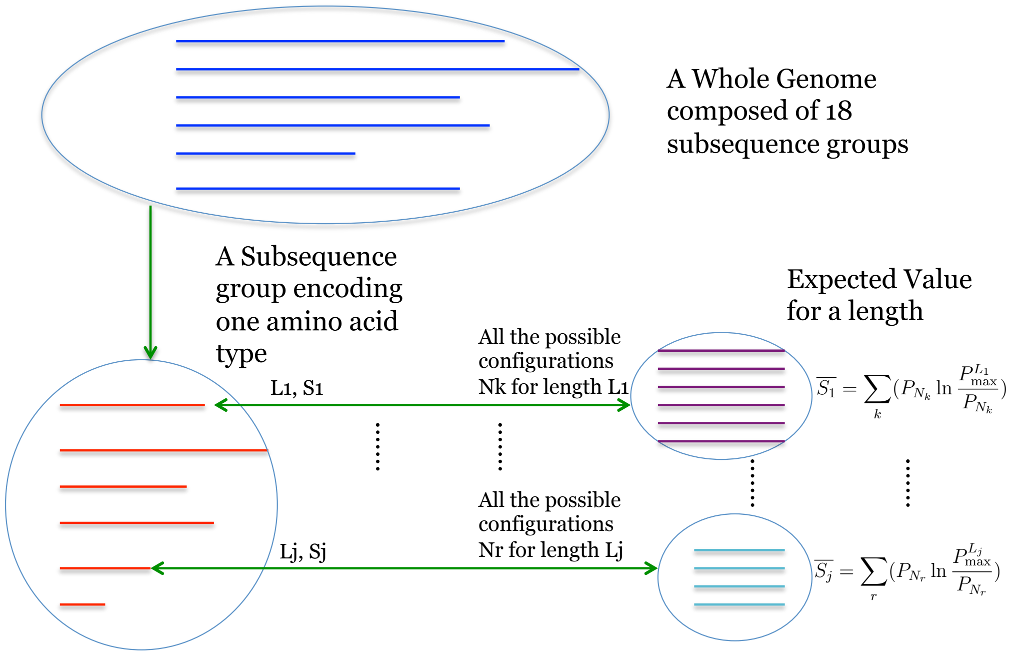

A Whole Genome composed of 18 subsequence groups
A Subsequence group encoding one amino acid type
Expected Value for a length
L1, S1
All the possible configurations Nk for length L1
$\overline{S_1} = \sum_k (P_{N_k} \ln \frac{P_{\max}^{L_1}}{P_{N_k}})$
Lj, Sj
All the possible configurations Nr for length Lj
$\overline{S_j} = \sum_r (P_{N_r} \ln \frac{P_{\max}^{L_j}}{P_{N_r}})$

Figure 6: Expected $Sn$ calculation. Theoretically speaking, codon sequences could be analyzed directly based on transcriptoms. However in this work, the downloaded genomic sequences only include the codon coding DNAs, therefore we perform operations on transcriptom-equivalent genomes based on watson crick base pairing. A whole genome is divided into 18 subsequence groups, each of which encodes one amino acid type. Assuming in the plotted subsequence group the $j$-th subsequence has the length $L_j$ and $S_j$. For the $j$-th subsequence, there is a corresponding $\overline{S}_j$. $\overline{S}_j$ depends on all the possible subsequence configurations $N_r$ for length $L_j$.

(EBI, UK) and the DNA Databank of Japan (DDBJ, Japan).

At the time of data preparation, ENSEMBL had available 811 genomes from 523 species in the fungi kingdom, 189 genomes from 119 species in the protist kingdom, and 44,046 genomes from 8244 species in the bacteria kingdom. In order to cover species with a wide phylogenetic range within a kingdom and diminish undesired sample aggregation influences, we selected species based on the level of taxonomy of phyla (phylum is the first taxonomic rank below kingdom) (Hibbett et al. (2007), Woese, Kandler and Wheelis (1990), Adl et al. (2012)). Take the bacteria kingdom for example, we retrieved all the available species names below each of the 28 major phyla of the bacteria kingdom ($N_i$ is the available sample

Table 3: Genomes from Fungi Kingdom

| Phylum | Available Genome Amounts | Selected Genome Amounts |
|---|---|---|
| Blastocladiomycota | 2 | 2 |
| Chytridiomycota | 10 | 10 |
| Glomeromycota | 8 | 8 |
| Microsporidia | 30 | 30 |
| Neocallimastigomycota | 4 | 4 |
| Ascomycota | 725 | 195 |
| Basidiomycota | 212 | 194 |
| Entomophthoromycotina | 2 | 2 |
| Mucoromycotina | 17 | 17 |

size of the $i$-th phylum, $i \in [1, 28]$). To enhance the comparability between kingdoms, we predefined the optimal overall sample size $N_{opt}$ to be between 400 to 500, meaning the optimal sample size for each of the 28 bacterial phyla ($N_p^{opt}$) is approximately $N_{opt}/28$. If the available $N_i$ is smaller than $N_p^{opt}$, we reserve all the available species in the $i$-th phylum for investigation. If $N_i$ is larger than $N_p^{opt}$, we need to select $N_p^{opt}$ species for investigation. The whole species list of the $i$-th phylum directly retrieved from ENSEMBL rank in the alphabetic order in which way the phylogenetically close species tend to gather together. In order to avoid selected species aggregating in a narrow phylogenetic range, we selected species at the interval of $L_i^{span}$ through the whole species list of the $i$-th phylum. $L_i^{span}$ is an integer by rounding the ratio $N_i/N_p^{opt}$.

Finally we downloaded locally the CDS (protein-coding sequence) FASTA files for all the selected species. In total we chose 462 genomes from the Fungi kingdom (release 36 in August 2017), 441 genomes in Bacteria kingdom (release 40 in July 2018), and 143 genomes in Protista kingdom (release 40 in July 2018). All the investigated species are listed in the appendix 'speciesList.xlsx'. The amounts of available genomes and selected genomes for all the 3 kingdoms are listed in detail in Table 3, Table 4 and Table 5.

Based on downloaded protein coding sequences, we generated valid codon sequences. For a clearer explanation, we take species *Saccharomyces cerevisiae* as an example. The original FASTA sequences begin with a single line description followed by lines of protein-coding DNA data, showing as follows:

Table 4: Genomes from Bacteria Kingdom

| Phylum | Available Genome Amounts | Selected Genome Amounts |
|---|---|---|
| Acidobacteria | 2 | 2 |
| Aquificae | 18 | 18 |
| Armatimonadetes | 5 | 5 |
| Bacteroidetes | 815 | 28 |
| Caldiserica | 1 | 1 |
| Chlamydiae | 228 | 28 |
| Chlorobi | 16 | 16 |
| Chloroflexi | 9 | 9 |
| Chrysiogenetes | 1 | 1 |
| Cyanobacteria | 228 | 28 |
| Deferribacteres | 5 | 5 |
| Deinococcus Thermus | 39 | 28 |
| Dictyoglomi | 2 | 2 |
| Elusimicrobia | 1 | 1 |
| Fibrobacteres | 2 | 2 |
| Firmicutes | 15650 | 28 |
| Fusobacteria | 76 | 28 |
| Gemmatimonadetes | 3 | 3 |
| Lentisphaerae | 1 | 1 |
| Nitrospirae | 17 | 17 |
| Planctomycetes | 21 | 21 |
| Proteobacteria | 19212 | 28 |
| Spirochaetes | 332 | 28 |
| Synergistetes | 17 | 17 |
| Tenericutes | 192 | 28 |
| Thermodesulfobacteria | 6 | 6 |
| Thermotogae | 40 | 28 |
| Verrucomicrobia | 24 | 24 |

Table 5: Genomes from Protist Kingdom

| Phylum | Available Genome Amounts | Selected Genome Amounts |
|---|---|---|
| Rhodophyta | 3 | 3 |
| Stramenopiles | 38 | 38 |
| Alveolata | 90 | 58 |
| Rhizaria | 6 | 6 |
| Euglenozoa | 20 | 20 |
| Amoebozoa | 15 | 15 |
| Apusozoa | 1 | 1 |
| Choanozoa | 2 | 2 |

```
> YHR055Ccdschromosome : R64 − 1 − 1...[Source : SGD; Acc : S000001097]
ATGTTCAGC...GGGAAATGA
> YPR161Ccdschromosome : R64 − 1 − 1 : ...[Source : SGD; Acc : S000006365]
ATGAGTGATAAT...GATCTATATTAG
......
```

As shown in the above example, we transformed protein-coding gene sequences into codon sequences for the codon level analysis, by way of grouping adjacent nucleotide triplets sequentially from the start codon to stop codon. If the gene length between the start codon and the stop codon is not a multiple of 3, it conveys the codon sequence contains wrong Open Reading Frame. We deleted such genes from the raw genome. The number of genes under investigation and excluded are listed as Table 6.

Table 6: Genes excluded from our analysis

| Kingdom | Total Gene Number | Excluded Gene Number |
|---|---|---|
| Fungi | 4554328 | 35748 |
| Bacteria | 1286467 | 6384 |
| Protist | 1439975 | 25142 |

We processed the original contents in the above example (the CDS FASTA file of *Saccharomyces cerevisiae*) into the format as a single line of gene name followed by a single line of codon sequences, which is convenient to retrieve gene name and process sequence data at the codon level :

Table 7: Format of Datasets for Codon Occurrence Configuration

| geneNeme | GAG | GAA | sublength |
|---|---|---|---|
| $> YHR055C$ | 1 | 5 | 6 |
| $> YPR161C$ | 14 | 23 | 37 |
| $> YOL138C$ | 24 | 59 | 83 |
| ... | ... | ... | ... |

YHR055C;

'ATG' 'TTC' 'AGC'...'GGG' 'AAA' 'TGA'

YPR161C;

'ATG' 'AGT' 'GAT' 'AAT'...'GAT' 'CTA' 'TAT' 'TAG'

......

All the genes are reserved as the valid codon sequences for the species of *Saccharomyces cerevisiae* after genome clearance procedure.

### 3.2.2 Generating Datasets of *Codon Occurrence Configurations*

Next we process the amino acid sequences. We split every gene into 18 subsequences each of which encodes one amino acid type. For each gene, we then obtained 18 codon occurrence configurations corresponding to 18 different subsequences individually.

Take the genome of the fungal species *Saccharomyces cerevisiae* as an example, codon occurrence configurations of subsequences encoding amino acid GLU in each gene are produced in the following format in Table 7. Information for each gene corresponds to an individual line. Each line includes gene name, number of copies of each synonymous codon type, and the subsequence length.

For example in Table 7 the row '$> YHR055C$ 1 5 6' means that the gene 'YHR055C' has 1 codon of 'GAA', 5 codons of 'GAG', and hence the subsequence coding amino acid GLU in the gene 'YHR055C' has the length of 6. Similarly the row '$> YPR161C$ 14 23 37' means that the gene 'YHR055C' has 14 codons of 'GAG' and 23 codons of 'GAA' , and hence the subsequence coding GLU in the gene YHR055C has the length of 37.

Matlab code to produce the codon occurrence configuration for each gene of all the genomes from Fungi kingdom is generalised as follows:

```
{% calculate and write syno ratio for each gene within genome
% set the synonymous codon usage table as follows
ctE={'GAG','GAA'};   %%Glu
ctH={'CAT','CAC'};   %%His
ctQ={'CAG','CAA'};   %%Gln
ctF={'TTT','TTC'}; %%Phe
ctY={'TAT','TAC'};     %%Tyr
ctC={'TGT','TGC'};   %%Cys
ctN={'AAT','AAC'};     %%Asn
ctK={'AAG','AAA'};     %%Lys
ctD={'GAT','GAC'};   %%Asp
ctI={'ATA','ATT','ATC'}; %%Ile
ctP={'CCG','CCA','CCT','CCC'};   %%Pro
ctT={'ACG','ACA','ACT','ACC'};   %%Thr
ctA={'GCG','GCA','GCT','GCC'}; %%Ala
ctV={'GTG','GTA','GTT','GTC'}; %%Val
ctG={'GGG','GGA','GGT','GGC'}; %%Gly
ctL={'TTG','TTA','CTG','CTA','CTT','CTC'};   %%Leu
ctS={'AGT','AGC','TCG','TCA','TCT','TCC'};     %%Ser
ctR={'AGG','AGA','CGG','CGA','CGT','CGC'};     %%Arg

text={ctE,ctH,ctQ,ctF,ctY,ctC,ctN,ctK,ctD,ctI,ctP,ctT,ctA,ctV,
    ctG,ctL,ctS,ctR};

fileName0='fungiNameList.csv';
fileID0=fopen(fileName0,'r');
speciesNamep=textscan(fileID0,'%s','Delimiter','\n');
speciesName=speciesNamep{1,1};
fclose(fileID0);

for tP=1:length(speciesName)
    fileName=[speciesName{tP},'.fa'];%%%fungi
    geneName=getSequenceName(fileName);
    pasteCodon=getCodonSequence(fileName);

    fmt1=('%s,%u,%u,%u\n');
    fmt2=('%s,%u,%u,%u,%u\n');
    fmt3=('%s,%u,%u,%u,%u,%u\n');
    fmt4=('%s,%u,%u,%u,%u,%u,%u,%u\n');

    fileIDe=fopen([speciesName{tP},'GluEratioFg.txt'],'a');
    fprintf(fileIDe,'%s,%s,%s,%s\n','geneNeme','GAG','GAA','sublength');

    fileIDh=fopen([speciesName{tP},'HisHratioFg.txt'],'a');
    fprintf(fileIDh,'%s,%s,%s,%s\n','geneNeme','CAT','CAC','sublength');

   ...... % set fileID for all the 18 amino acids
                fileList={fileIDe,fileIDh,fileIDq,fileIDf,fileIDy,fileIDc,fileIDn,fileIDk,fileIDd,fileIDi,fileIDp,fileIDt,fileIDa,fileIDv,fileIDg,fileIDl,fileIDs,fileIDr};
    fmList={fmt1,fmt1,fmt1,fmt1,fmt1,fmt1,fmt1,fmt1,fmt1,fmt2,fmt3,fmt3,fmt3,fmt3,fmt3,fmt4,fmt4,fmt4};

    for CTcount=1:18

        synoEle=text{CTcount}; %%locate synonymous codon family

        for countCd=1:length(pasteCodon)

            for synoCount=1:length(synoEle)
                    SYNO(synoCount)=length(find(ismember(pasteCodon{countCd},synoEle{synoCount})));
            end

            fprintf(fileList{CTcount},fmList{CTcount},geneName{countCd},SYNO,sum(SYNO));
```

```

          end

          fclose(fileList{CTcount});

     end


end
}
```

### 3.2.3 Preparing Global Codon Usage Table

Adopting codon occurrence configuration N of each subsequence (symbols refer to Table 2), we generated global codon usage table for each species of a kingdom. Global codon usage pattern of a species is obtained by adding up occurrences of each type of codon dispersed in genes throughout the whole genome, namely consider the genome as a very long sequence, and the global codon usage pattern of one synonymous codon family is the codon occurrence configuration corresponding to the whole genome sequence $N_w = [n_1^w, n_2^w, ...n_{|C^{AA}|}^w]$, where $\sum_{i=1}^{|C^{AA}|} n_i^w = L_{AA}^w$, $L_{AA}^w$ is the codon sequence length encoding amino acid $AA$ within the whole genome. The global codon usage table contains species names in the first column, followed by columns for synonymous codon usage ratios for all the synonymous codon families. Global codon usage ratio $\rho_j^i$ of the $i$-th synonymous codon in the $j$-th synonymous codon family is calculated as $\rho_j^i = \frac{n_i^w}{\sum_{i=1}^{|C_j^{AA}|} n_i^w}$, where $C_j^{AA}$ is the $j$-th synonymous codon family, $i \in [1, C_j^{AA}]$, $j \in [1, 18]$.

For example, the global codon usage of amino acid GLU(E) in fungal species *Candida auris* is produced with the following format:

speciesName, E(GAG, GAA)
*candida_auris*, 0.5587102719662976, 0.4412897280337024

This means that in the species *Candida auris*, at the whole genome level to encode amino acid GLU(E), the global codon usage of codon 'GAG' and codon

'GAA' has the ratio of 0.5587102719662976 over 0.4412897280337024.

### 3.2.4 Datasets Prepared For $Sn$ Calculation

We have introduced that $Sn$ can be derived from three terms: the multinomial distribution probability of a subsequence $P_N^{AA,g}$, the maximum probability corresponding to the subsequence length $P_{\max}^{AA,L}$, and the subsequence length $L$. Therefore to obtain $Sn$ values we prepared datasets containing $P_N^{AA,g}$, $P_{\max}^{AA,L}$ and $L$ for each subsequence within genes throughout the whole genome.

Take the species *Saccharomyces cerevisiae* as an example, the obtained datasets for $Sn$ calculation is shown in Table 8:

Table 8: Format of Datasets for $Sn$ Calculation

| AA | $L^{AA,g}$ | $P_N^{AA,g}$ | $P_{\max}^{AA,L}$ | GeneId | |
|---|---|---|---|---|---|
| E | 1 | 5.000000e-01 | 5.000000e-01 | 10 | |
| E | 1 | 5.000000e-01 | 5.000000e-01 | 16 | |
| ... | ... | ... | ... | ... | ... |
| E | 2 | 5.000000e-01 | 5.000000e-01 | 30 | |
| E | 2 | 2.500000e-01 | 5.000000e-01 | 52 | |
| ... | ... | ... | ... | ... | ... |
| E | 4 | 3.750000e-01 | 3.750000e-01 | 476 | |
| ... | ... | ... | ... | ... | ... |
| E | 316 | 9.203490e-09 | 4.484902e-02 | 156 | |
| E | 435 | 3.331582e-26 | 3.818984e-02 | 4881 | |
| H | 1 | 5.000000e-01 | 5.000000e-01 | 1 | |
| H | 1 | 5.000000e-01 | 5.000000e-01 | 6 | |
| ... | ... | ... | ... | ... | ... |

This table contains information about all the subsequences (with different lengths) encoding 18 different amino acids within each gene throughout the whole genome. Symbols in the header sequentially represent 'amino acid type', 'subsequence length', 'multinomial distribution probability to observe the codon occurrence configuration for such subsequence', 'maximum multinomial distribution probability of such subsequence length', 'gene index within the genome'

For example the line '$\texttt{E}, 2, 2.500000\texttt{e} - 01, 1, 5.000000\texttt{e} - 01, 52$' means:

(1) The index for this gene is 52. The 52nd gene in the genome has 2 codons to encode amino acid GLU(E), namely the length of the subsequence to code E in this gene is 2;

(2) Assuming the global underlying codon usage probability is [1/2, 1/2], then the probability to observe the codon occurrence configuration of the subsequence encoding GLU in the 52nd gene is 2.500000e-01.

(3) Among all the $P_N^{AA,g}$ for subsequence of length 2 encoding amino acid GLU, the maximum probability to observe the most likely subsequence is 5.000000e-01.

(4) Based on the $P_N^{AA,g}$, $P_{\max}^{AA,L}$ and $L$, each gene obtains 18 corresponding $Sn^{AA,g}$ values for 18 different amino acids: $Sn^{AA,g} = \left(\ln \frac{P_{\max}^{AA,L}}{P_N^{AA,g}}\right)/L$.

### 3.2.5 Two Types of Control Genomes

We produced two types of control genomes to compare with the actual genome data. One control group is the equally substituted artificial genome where each codon is replaced by synonymous codons with the same probability $|C^{AA}|/L^{AA,g}$. The other control group is biased substituted artificial genome where each codon is replaced by synonymous codons with the weighted probability according to a global codon usage preference. Again take the species *Candida auris* global codon usage 'GAG' and 'GAA' encoding amino acid GLU as an example. For the equally substituted genome, we chose codon 'GAG' with the probability of 0.5, and chose codon 'GAA' also with the probability of 0.5. For biased substituted genome, global codon usage table for amino acid GLU in species *Candida auris* is [0.5587, 0.4412], thence we chose codon 'GAG' with the probability of 0.5587, while chose codon 'GAA' with the probability of 0.4412.

Substituted genomes are created as below in detail:

(1) Equally replaced codon sequence: each codon is replaced by synonymous codons with the same probability. For example for the subsequence of length 10 and encoding amino acid isoleucine (Ile) which is encoded by 3 synonymous codons ATA, ATT, ATC, we randomly sample 10 integers $D_i$ under a discrete uniform distribution, where $D_i$ has the value of 1, 2 or 3, $i \in [1, 10]$, 1,2,3 represent synonymous codons ATA, ATT, ATC individually. If sampled number is 1, we choose ATA for the codon position, if 2 we choose ATT, and if 3 we choose ATC.

(2) Biased replaced codon sequence: each codon is replaced by its synonymous codons with a weighted probability according to the global codon usage table. To construct such control group: firstly find synonymous codon usage ratio in the global codon usage table for an interested genome. Secondly replace each codon within the genome with the synonymous codons with the weighted probability according to the found synonymous codon usage ratio. Take subsequences encoding Ile of length 10 for example, if the global codon usage ratio among ATA, ATT, ATC is 0.1:0.3:0.6, we randomly sample 10 decimals $D_i$ under the continuous uniform distribution, where $D_i \in [0, 1]$, $i \in$ [1,10] . If $D_i \leq 0.1$ we choose ATA for the codon position, if $0.1 < D_i \leq 0.4$ we choose ATT, otherwise we choose ATC. When artificial genome is generated, we perform the same calculation as the real observed genome.

### 3.2.6 Three Types of Datasets for $Sn$ Calculation

Datasets containing $P_N^{AA,g}$, $P_{\max}^{AA,L}$ and $L$ are generated based on real and substituted genomes. As introduced in section 3.1.3, multinomial distribution probability $P_N^{AA,g}$ depends on two parameters which are the underlying global codon usage probability $P^{AA}$ and the codon occurrence configuration $N$ of the subsequence.

There is no universal rule defining what the underlying codon usage probability $P^{AA}$ should be. $Sn$ as the proposed measure of CUB, as long as each subsequence is assessed under the same assumption and consistent standard, the values of $Sn$ of different subsequences should be comparable and meaningful. Thence we assume $P^{AA}$ ($P^{AA} = [P_1, P_2, ...P_i, ...P_m], i \in [1, m]$) has the equal entry $1/|C^{AA}|$ for convenience, for example for the 3 synonymous codon family, the underlying codon usage probability for each synonymous codon is 1/3.

In this work, we will analyse three different datasets. Firstly, the dataset of $Sn$ values derived from the actual genomes. We call this the dataset 'T'. We will then compare this with two different control datasets. These have been generated so as to implement two different hypotheses about the CUB.

To generate these control datasets we created two artificial genomes (detailed procedures refer to section 3.2.5 'Two Types of Control Genomes'). The first one consists of the same sequences that we consider in the original dataset, but we replaced each codon by a random codon. By construction, this dataset has no codon usage bias and should follow the multinomial distribution with all synonymous codons being equally chosen. The second artificial genome is also an artificial version of the actual sequences, and again we replaced all codons by random ones, but now the random substitution was done such that the codon replacement reflected the empirically determined global codon usage bias of the species. By construction, this latter dataset implements a codon usage bias that follows the multinomial distribution with unequal underlying synonymous codon usage.

From these two artificial datasets we then calculated the $Sn$ values as we did for the real genomes, thus obtaining two control datasets 'Ta' and 'Tba' which we could compare against the dataset 'T'.

Three types of datasets prepared for $Sn$ calculation are displayed as Table 9, 'T' represents 'table', 'a' represents 'artificial', 'b' represents 'biased'):

Table 9: Datasets for $Sn$ Calculation

| Dataset | Genome Type | $P^{AA}$ |
|---|---|---|
| T | Real Genome | Equal Synonymous Codon Usage |
| Ta | Equally Substituted Artificial Genome | Equal Synonymous Codon Usage |
| Tba | Biased Substituted Artificial Genome | Equal Synonymous Codon Usage |

(1) T: $P_N^{AA,g}$ are calculated based on the real genome, and the underlying codon usage probability $P^{AA}$ has equal entries, to be specific $P^{AA} = [P_1, P_2, ..., P_m]$ where $P_1 = P_2 = ... = P_M$.

(2) Ta: $P_N^{AA,g}$ are calculated based on the artificial genome, and the underlying codon usage probability $P^{AA}$ has equal entries. The artificial genome is constructed by replacing codon with its synonymous codons with equal probability.

(3) Tba: $P_N^{AA,g}$ are calculated based on the artificial genome, and the underlying codon usage probability $P^{AA}$ has equal entries. The artificial genome is constructed by replacing codon with its synonymous codons with biased probability according to global codon usage table.

### 3.2.7 Expected $Sn$ Dataset

$\overline{Sn}$ is the expected values of $Sn$ for subsequences of the same length. We prepared codon occurrence configuration tables for 2, 3, 4, 6 synonymous codon family of lengths from 1 to 400, and based on the configuration tables we further prepared $\overline{Sn}$ table for lengths from 1 to 400.

For each length we generated a table which contains all the possible $N_j^{AA,L}$ and corresponding $Sn^j$, based on which we obtain $\overline{Sn}$.

For example, for 2 synonymous codon family, we prepared 400 tables, each table corresponds to one length. For the table of length=2, it displays as the following:

```
configuration, Sn
[0, 2], 0.3465735902799726
[2, 0], 0.3465735902799726
[1, 1], 0
```

$\overline{Sn} = 0.1732867951399863$ for 2 synonymous codon family of length 2. It is calculated as $\overline{Sn} = \sum_j P_{N_j} Sn^j = 0.25 \cdot 0.3465735902799726 + 0.25 \cdot 0.3465735902799726 + 0.5 \cdot 0 = 0.1732867951399863$

In summary for CUB measure, we prepared (1) Codon occurrences configuration datasets; (2) Global codon usage table for all the species in 3 kingdoms; (3) $Sn$ datasets for both observed genome and substituted genome under the assumption of uniform $P^{AA}$; (4) $\overline{Sn}$ datasets.

## 3.3 Hypothesis Test Results for $Sn$ Across Species

We applied hypothesis test for $Sn$ to each gene in species *S.cerevisiae* adopting the method introduced in section 3.1.5 (Statistical Power of $Sn$). There are 6692 genes in *S.cerevisiae*, among which we count the genes for which $Sn$ have the statistical power to indicate that the observed distribution of codon sequences are significantly different from the theoretical multinomial distribution of codon sequence when there is no CUB (significance level $\alpha = 0.05$). Finally across the whole genome we calculate the proportion of the genes which are significantly different from cases of no CUB, as shown in Table 10.

Table 10: Hypothesis Test for *Sn* in *S.cerevisiae*

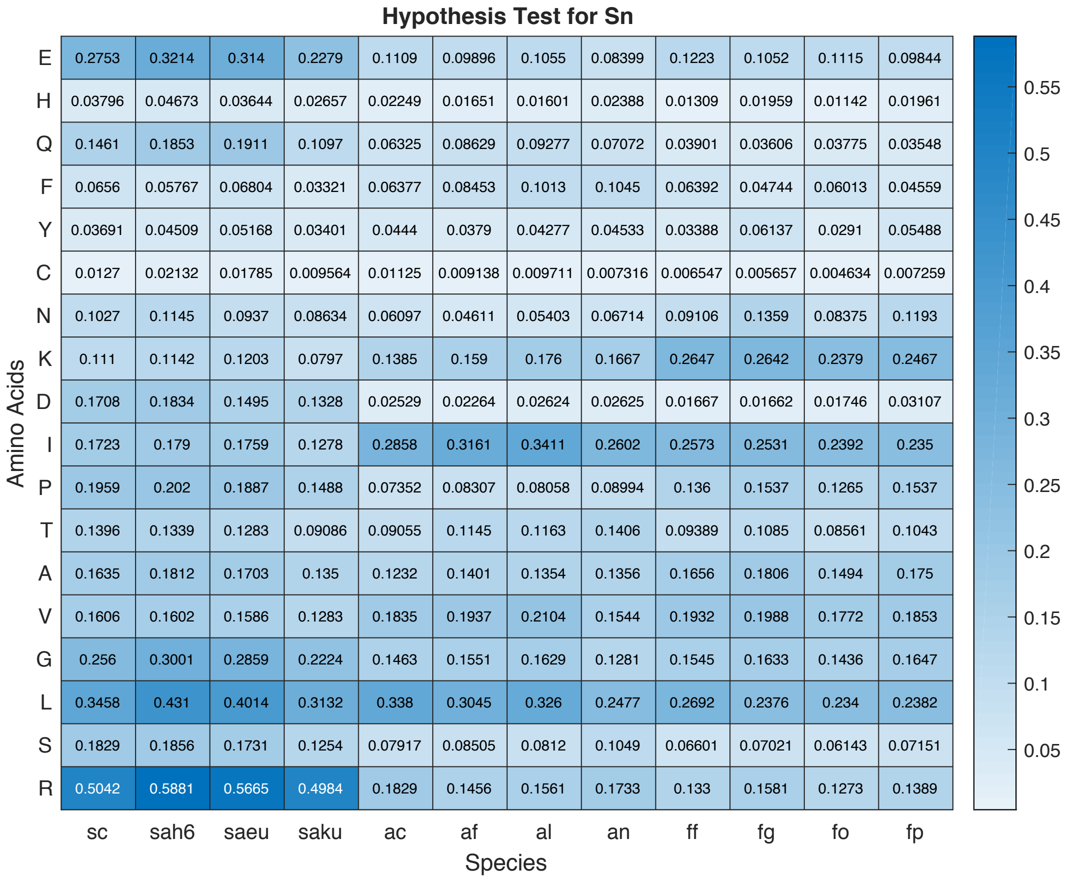
| Amino Acid | Reject H0 | Accept H0 | Proportion of Genes Rejecting H0 |
|---|---|---|---|
| E | 1842 | 4850 | 2.752540e-01 |
| H | 254 | 6438 | 3.795577e-02 |
| Q | 978 | 5714 | 1.461447e-01 |
| F | 439 | 6253 | 6.560072e-02 |
| Y | 247 | 6445 | 3.690974e-02 |
| C | 85 | 6607 | 1.270173e-02 |
| N | 687 | 6005 | 1.026599e-01 |
| K | 743 | 5949 | 1.110281e-01 |
| D | 1143 | 5549 | 1.708010e-01 |
| I | 1153 | 5539 | 1.722953e-01 |
| P | 1311 | 5381 | 1.959056e-01 |
| T | 934 | 5758 | 1.395696e-01 |
| A | 1094 | 5598 | 1.634788e-01 |
| V | 1075 | 5617 | 1.606396e-01 |
| G | 1713 | 4979 | 2.559773e-01 |
| L | 2314 | 4378 | 3.457860e-01 |
| S | 1224 | 5468 | 1.829050e-01 |
| R | 3374 | 3318 | 5.041841e-01 |

Null hypothesis H0 is: the observed codon sequence has the same multinomial distribution with the case with no codon usage bias. Rejection to H0 means that there exists codon usage bias in the observed codon sequence.

Next we apply the hypothesis test for *Sn* to 16 species (species name and abbreviation shown in Table 11), and obtained the result shown in Figure 7. There are clear difference between amino acids but also between species in which proportion of genes have significant codon usage bias.

## 3.4 Sequence Specific Measure Adopting *Sn* values

*Sn* measures CUB of a sequence encoding a particular type of amino acid. Because protein abundance is assumed to be an important driver of CUB, we adopt *Sn* to investigate relationships between protein abundance per cell and CUB in *S.cerevisiae*. 1341 genes in *S.cerevisiae* have available protein abundance data



| | sc | sah6 | saeu | saku | ac | af | al | an | ff | fg | fo | fp |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| E | 0.2753 | 0.3214 | 0.314 | 0.2279 | 0.1109 | 0.09896 | 0.1055 | 0.08399 | 0.1223 | 0.1052 | 0.1115 | 0.09844 |
| H | 0.03796 | 0.04673 | 0.03644 | 0.02657 | 0.02249 | 0.01651 | 0.01601 | 0.02388 | 0.01309 | 0.01959 | 0.01142 | 0.01961 |
| Q | 0.1461 | 0.1853 | 0.1911 | 0.1097 | 0.06325 | 0.08629 | 0.09277 | 0.07072 | 0.03901 | 0.03606 | 0.03775 | 0.03548 |
| F | 0.0656 | 0.05767 | 0.06804 | 0.03321 | 0.06377 | 0.08453 | 0.1013 | 0.1045 | 0.06392 | 0.04744 | 0.06013 | 0.04559 |
| Y | 0.03691 | 0.04509 | 0.05168 | 0.03401 | 0.0444 | 0.0379 | 0.04277 | 0.04533 | 0.03388 | 0.06137 | 0.0291 | 0.05488 |
| C | 0.0127 | 0.02132 | 0.01785 | 0.009564 | 0.01125 | 0.009138 | 0.009711 | 0.007316 | 0.006547 | 0.005657 | 0.004634 | 0.007259 |
| N | 0.1027 | 0.1145 | 0.0937 | 0.08634 | 0.06097 | 0.04611 | 0.05403 | 0.06714 | 0.09106 | 0.1359 | 0.08375 | 0.1193 |
| K | 0.111 | 0.1142 | 0.1203 | 0.0797 | 0.1385 | 0.159 | 0.176 | 0.1667 | 0.2647 | 0.2642 | 0.2379 | 0.2467 |
| D | 0.1708 | 0.1834 | 0.1495 | 0.1328 | 0.02529 | 0.02264 | 0.02624 | 0.02625 | 0.01667 | 0.01662 | 0.01746 | 0.03107 |
| I | 0.1723 | 0.179 | 0.1759 | 0.1278 | 0.2858 | 0.3161 | 0.3411 | 0.2602 | 0.2573 | 0.2531 | 0.2392 | 0.235 |
| P | 0.1959 | 0.202 | 0.1887 | 0.1488 | 0.07352 | 0.08307 | 0.08058 | 0.08994 | 0.136 | 0.1537 | 0.1265 | 0.1537 |
| T | 0.1396 | 0.1339 | 0.1283 | 0.09086 | 0.09055 | 0.1145 | 0.1163 | 0.1406 | 0.09389 | 0.1085 | 0.08561 | 0.1043 |
| A | 0.1635 | 0.1812 | 0.1703 | 0.135 | 0.1232 | 0.1401 | 0.1354 | 0.1356 | 0.1656 | 0.1806 | 0.1494 | 0.175 |
| V | 0.1606 | 0.1602 | 0.1586 | 0.1283 | 0.1835 | 0.1937 | 0.2104 | 0.1544 | 0.1932 | 0.1988 | 0.1772 | 0.1853 |
| G | 0.256 | 0.3001 | 0.2859 | 0.2224 | 0.1463 | 0.1551 | 0.1629 | 0.1281 | 0.1545 | 0.1633 | 0.1436 | 0.1647 |
| L | 0.3458 | 0.431 | 0.4014 | 0.3132 | 0.338 | 0.3045 | 0.326 | 0.2477 | 0.2692 | 0.2376 | 0.234 | 0.2382 |
| S | 0.1829 | 0.1856 | 0.1731 | 0.1254 | 0.07917 | 0.08505 | 0.0812 | 0.1049 | 0.06601 | 0.07021 | 0.06143 | 0.07151 |
| R | 0.5042 | 0.5881 | 0.5665 | 0.4984 | 0.1829 | 0.1456 | 0.1561 | 0.1733 | 0.133 | 0.1581 | 0.1273 | 0.1389 |

Figure 7: This heatmap summarises the hypothesis test results for Sn across 16 species where x axis displays the species name and y axis displays the amino acids. Each chess of the heatmap shows that among the whole genome of such species the proportion of $Sn$ which have statistical power to indicate that the values of $Sn$ imply the strength of CUB. Darker color suggests that larger proportion of genes in the species have codon usage bias.

Table 11: 16 Species for Hypothesis Test on $Sn$

| Species Name | Abbreviation |
|---|---|
| Saccharomyces cerevisiae | sc |
| Saccharomyces arboricola_$h$_6 | sah6 |
| Saccharomyces eubayanus | saeu |
| Saccharomyces kudriavzevii | saku |
| Aspergillus clavatus | ac |
| Aspergillus flavus | af |
| aspergillus lentulus | al |
| Aspergillus niger | an |
| Fusarium fujikuroi | ff |
| Fusarium graminearum | fg |
| Fusarium oxysporum | fo |
| Fusarium poae | fp |

from Protein Abundance Online Database: PaxDb, where each protein entity is enumerated relative to all other protein molecules in the cell. Compared to 'molar concentration' or 'molecules per cell' such way to describe protein abundance has the advantage of being independent of cell-size (Wang et al. (2012)). We analyse these 1341 genes aiming to explore relationships between protein abundance and CUB.

### 3.4.1 Relationship Analysis between Protein Abundance and Sequence-specific CUB Adopting $Sn$

Each gene is composed of 18 subsequences which correspond to 18 subsequence lengths and 18 $Sn$ values. The literature reports an established relationship between translation efficiency and codon usage. To validate $Sn$ as a measure of codon usage bias, we wished to explore whether this widely acknowledged relationship is also apparent in the $Sn$ values. In addition, an inverse relationship between length and expression levels has also been reported, and we therefore included gene length data in our analysis. For all the genes in *S.cerevisiae*, we plot $Sn$ values and subsequence lengths against protein abundances separately, aiming to find the relationship among $Sn$ values, subsequences lengths and protein abundances.

Results are shown in Figure 8 where we display 4 example plots of amino acids

Phe(F), Ile(I), Pro(P) and Arg(R) as the representatives for 18 amino acid types. There are consistent trends: in the low protein abundance region, subsequence lengths tend to reach the large values, whereas in the high protein abundance region subsequence lengths decrease. In contrast $Sn$ values show the opposite trend, strongly avoiding low $Sn$ values for highly expressed proteins. This conveys that gene length and CUB act together: high expression requires high $Sn$ and short length. In order to achieve high protein expression levels, genes cannot exceed a maximum size as well requiring a minimal $Sn$ value.

From the online database, the protein abundance distribution in *S.cerevisiae* is shown in Figure 9. To further testify our above conclusion derived from $Sn$ that gene length and CUB act together to satisfy high protein expression, we extracted genes from the lowest protein abundance region ($< 5$) and the highest abundance region ($>10000$) to form two groups, then plot their subsequence lengths against their corresponding $Sn$ values.

Results shown in Figure 10 confirm the consistent pattern with Figure 8 that is: (1) for the genes of the high protein abundance group, $Sn$ values tend to be high and meanwhile subsequence lengths tend to be short; (2) for genes of the low protein abundance group $Sn$ values tend to be low and subsequence lengths tend to be long. Our finding strongly supports the reported cases which state that highly expressed genes are more biased, and are shorter (Duret and Mouchiroud (1999), Moriyama and Powell (1998), Song et al. (2017)). Our way of treating the data comes to the same conclusions as previous evidences in literatures.

### 3.4.2 Application of $Sn$ in Homologous Genes

Next we apply $Sn$ measure in groups of genes which possess specific phylogenetic properties.

#### 3.4.2.1 Introduction of Homologous Genes

In biology homology means the existences who share ancestry between a pair of structures in different taxa (Hall (2007)). Species taxon is a term in classification, which is identified by taxonomists by observing from morphological, behavioural, or genetic aspects. *S.cerevisiae* at the taxonomy level of fungi kingdom is shown in Table 12.

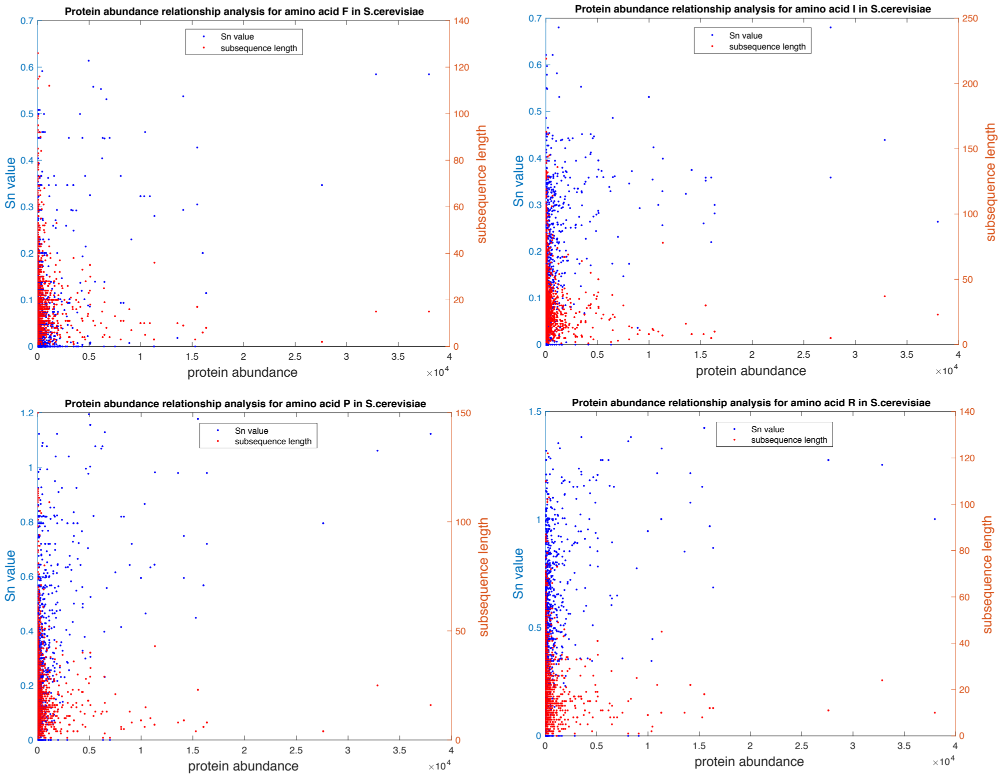


Figure 8: Sn values and subsequence length against protein abundance in *S.cerevisiae*. x axis represents the protein abundance. Azure left y axis is the gauge for Sn values. Orange right y axis is the gauge for subsequence length. Each Blue dot represents a variable pair of the protein abundance and its corresponding $Sn$ value. Each red dot represents a variable pair of the protein abundance and its corresponding subsequence length. In the low protein abundance regions, Sn distribution is random and subsequence reaches relative long length compared to the high protein abundance regions. In the high protein abundance region subsequence lengths are strikingly short but Sn values are distinguishably high.

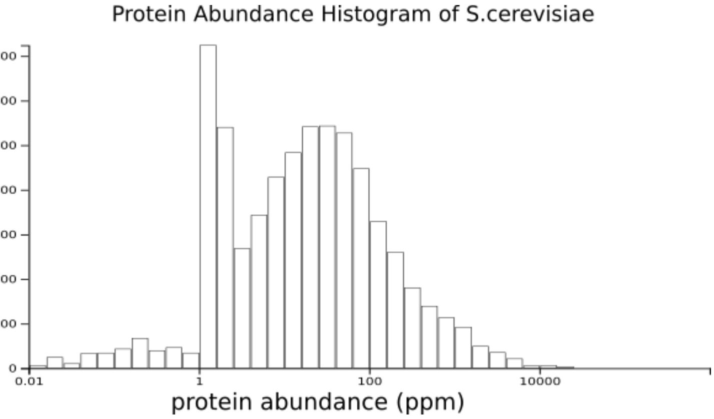


Figure 9: Protein abundance histogram in *S.cerevisiae*. Image resource: PaxDb: Protein Abundance Database

Table 12: Scientific Classification of *S.cerevisiae*

| | |
|---|---|
| Domain | Eukaryota |
| Kingdom | Fungi |
| Phylum | Ascomycota |
| Subphylum | Saccharomycotina |
| Class | Saccharomycetes |
| Order | Sacchromycetales |
| Family | Saccharomycetaceae |
| Genus | Saccharomyces |
| Species | *S.cerevisiae* |

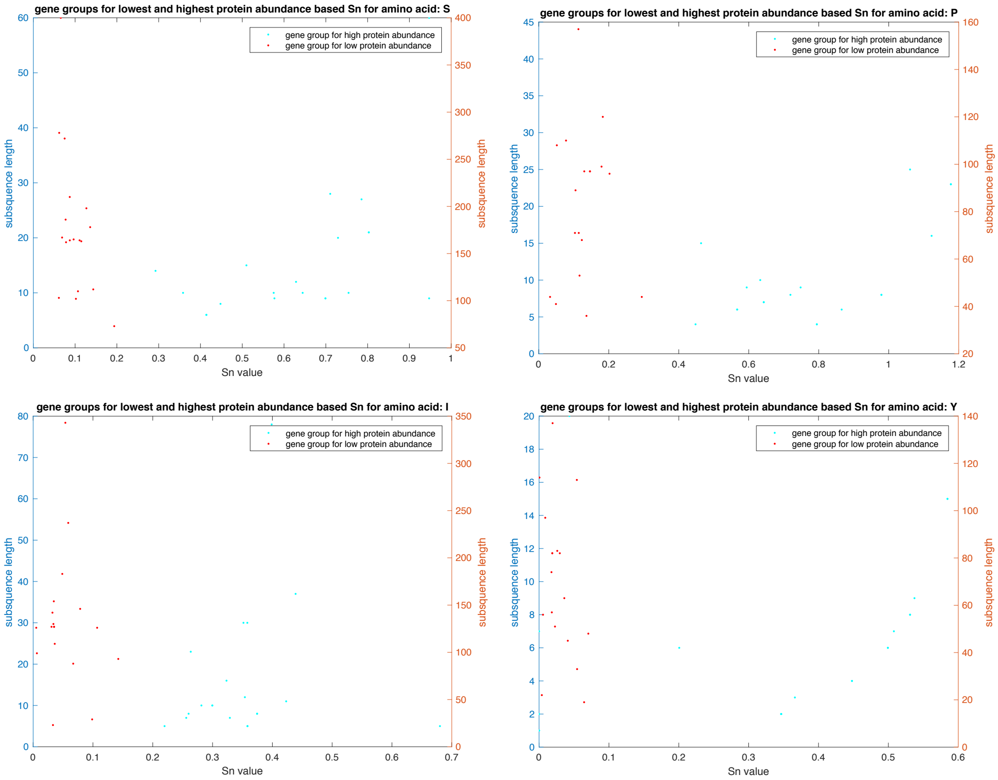


Figure 10: Subsequence lengths against Sn values. x axis represents $Sn$ values. Azure left y axis is the gauge for the subsequence lengths of genes in the high protein abundance region displayed as azure dots. Red right y axis is the gauge for the subsequence lengths of genes in the low protein abundance region displayed as red dots. Red dots aggregate in the $Sn$ regions of small values $< 0.2$ while Azure dots aggregate in the regions of short subsequence lengths. By contrary, red dots reach much longer length than blue dots, and blue dots reach much higher $Sn$ values than red dots.

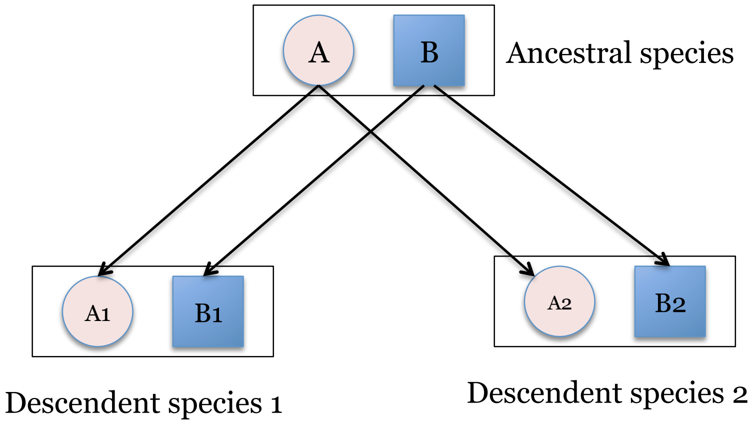


Figure 11: Ancestral species has two genes A and B which are paralogs. After speciation happens, ancestral species developed two descendent species. Within descendent species 1, gene A1 and gene B1 are paralogs; within descendent species 2, gene A2 and gene B2 are paralogs. Meanwhile A1 and A2 are orthologs, B1 and B2 are orthologs, A1 and B2 are orthologs, A2 and B1 are orthologs.

When it comes to homology among gene sequences, we define homologies according to sequence alignments technique, which typically infer DNA homology according to the sequence similarity, and significant similarity offers strong evidence that two gene sequences are related by divergent evolution from a common ancestor (Koonin and Galperin (2013)). Sequence alignment include various algorithms, we will introduce in general how Ensemble online database defines homolgos in the next subsection.

Two segments of DNA have shared ancestry because of either a speciation event (orthologs) or a duplication event (paralogs). Orthologs exist in different species by vertical descent from a single gene of the last common ancestor, while paralogs are duplicated genes occupying different positions in the same genome (Koonin (2005)). Meanings of Orthologs and paralogs are depicted in the Figure 11.

Homologous genes have specific phylogenetic relationships and hence are useful objects for CUB analysis.

#### 3.4.2.2 Retrieving Homologs from Ensemble Database

Homologs for all genes within *S.cerevisiae* among 462 species can be retrieved from ENSEMBL online database.

Ensemble homology database use systematic sequence comparison approaches to generate homology information. To be specific the pipeline has 7 basic steps:

Step 1: Load annotated sequences in Ensembl Genome Database.

Step 2: Run NCBI Blast between each paired genes (include self) in a genome-wise manner, and the measure of sequence similarity adopted by NCBI blast is E-value (Pearson and Lipman (1988)).

Step 3: Generate gene hierarchical clusters based on Blast results, where distance among genes are calculated based on E-values.

Step 4: Hierarchical clusters are split recursively and each recursive split is to find a branch roughly holds half of the nodes. Splitting stops until each cluster size is below a predefined cluster size (Howe, Bateman and Durbin (2002)).

Step 5: Perform multiple protein sequences alignment within each cluster. Multiple alignment is a sequence alignment technic for three or more sequences which are assumed to have an evolutionary relationship.

Step 6: Build a phylogenetic tree based on protein multiple alignment results. This complex procedure produces different trees adopting different algorithms then uses 'tree merging' algorithm to generate a consensus 'protein' phylogenetic tree. The purpose to adopt different algorithms is to take into account of species tree topology based on the NCBI taxonomy and also the results from multiple protein sequences alignment.

Step 7: From each gene tree, infer gene pairwise relations of ortholog and paralogy types.

By taking the above steps, homologous genes produced by Ensembl takes into account of gene sequence similarity, codon sequence similarity and species taxonomy.

MATLAB is capable to abstract homology information from Ensembl database for any given gene via Ensembl's Representation State Transfer (REST) API. Code is shown as follows, which produces the lists 'homoSpecies' and 'homoGene' special for the target gene 'stringName':

```
function [homoSpecies,homoGene,l] = getHomoInfor(stringName)

weblink=['http://rest.ensemblgenomes.org/homology/id/',
    stringName,'...
?compara=fungi&content-type=application/json&sequence=cdna&type
    ...
=orthologues&format=condensed'];
option=weboptions('Timeout',120);
try
strut=webread(weblink,option);

l=length(strut.data.homologies);
homoSpecies=cell(1,1);
homoGene=cell(1,1);

for fdID=1:l
    homoSpecies{fdID}=strut.data.homologies(fdID).species;
    homoGene{fdID}=strut.data.homologies(fdID).id;
end

catch          % avoid computing pausing
    fid=fopen('webreadFail.txt','a');  %store webread failure
        species
    fprintf(fid,'%s,',stringName);
    fclose(fid);
    homoSpecies={}; %if webread fail return empty value to
        avoid error
    homoGene={};
    l=0;

end

end
```

#### 3.4.2.3 CUB Patterns in Orthologs

For each gene in *S.cerevisiae*, we retrieved orthologous information among 461 species in fungi kingdom. Based on retrieved 'homoSpecies' and 'homoGene', we searched our prepared *Sn* datasets of orthologs for 5818 genes in *S.cerevisiae.*

We studied 50 genes grouped according to their function as Table 13 and

Table 13: Grouped 30 genes

| Ribosomal Protein | Transcription Factor (DNA binding) | Glucose Metabolic Process |
|---|---|---|
| RPL28 YGL103W | ACM1 YPL267W | GLK1 YCL040W |
| RPL25 YOL127W | AIM20 YIL158W | HXK1 YFR053C |
| RPL10 YLR075W | BIK1 YCL029C | PKP2 YGL059W |
| RPL32 YBL092W | BNI5 YNL166C | HXK2 YGL253W |
| RPL3 YOR063W | BUB3 YOR026W | TDH3 YGR192C |
| RPP0 YLR340W | BUD3 YCL014W | DOG1 YHR044C |
| RPS12 YOR369C | CBF5 YLR175W | FBP26 YJL155C |
| RPS13 YDR064W | CDC10 YCR002C | PGM1 YKL127W |
| RPS15 YOL040C | CDC123 YLR215C | PRM15 YMR278W |
| RPS2 YGL123W | CDC16 YKL022C | ZWF1 YNL241C |

Table 14. Based on $Sn$ values derived from the real genome and artificial genome without any bias, we obtain the result displayed as Figure 12.

Table 14: Grouped 20 genes

| Amino Acid Biosynthetic Process | Cell Cycle |
|---|---|
| ACO2 YJL200C | CDC14 YFR028C |
| ADE3 YGR204W | YCS4 YLR272C |
| CYS3 YAL012W | CKS1 YBR135W |
| HIS5 YIL116W | CLN1 YMR199W |
| ILV2 YMR108W | CLN3 YAL040C |
| LEU3 YLR451W | TPK1 YJL164C |
| LYS20 YDL182W | DOC1 YGL240W |
| MET2 YNL277W | YOX1 YML027W |
| TRP1 YDR007W | SPC19 YDR201W |
| PRO2 YOR323C | SSD1 YDR293C |

In Figure 12 we displayed heatmaps based on $Sn$ values for subsequences encoding amino acids Ile, Asp, Gly, and Arg within 50 genes in *S.cerevisiae*. The four amino acids are randomly chosen to display from 2, 3, 4, 6 synonymous codon families. The real codon sequences universally have darker colors compared to the random replaced artificial codon sequences, which illustrates that real genes broadly have higher $Sn$ values than artificial ones without any bias. Differences between real and artificial genes become stronger when more choices of synonymous codons are available to encode the amino acid. In addition seen from the perspective of the gene functional groups, genes in the cell cycle group display the slightest differences between the real and artificial genomes, which indicates that the genes have less codon usage bias in this group than other functional groups.

Hitherto we have introduced $Sn$ as an amino acid specific and sequence specific measure of CUB. When we applied $Sn$ to real sequences in *S.cerevisiae*, we found that with the increasing demand of protein in cell genes decrease their lengths and guarantee high level of CUB. When we applied $Sn$ to the homology analysis, we found that CUB is stronger in real sequences than artificial ones, and also stronger in bigger sized synonymous codon families, in addition CUB patterns are related to gene functions.

If quantifying CUB at the whole genome level with one measure which includes CUB information of all the subsequences and 18 amino acids, $Sn$ needs to be combined in a validated way, which tackles with different subsequence lengths and 18 amino acids throughout the genome. Next chapter we will propose the method to combine $Sn$ for the genome-wide CUB analysis.

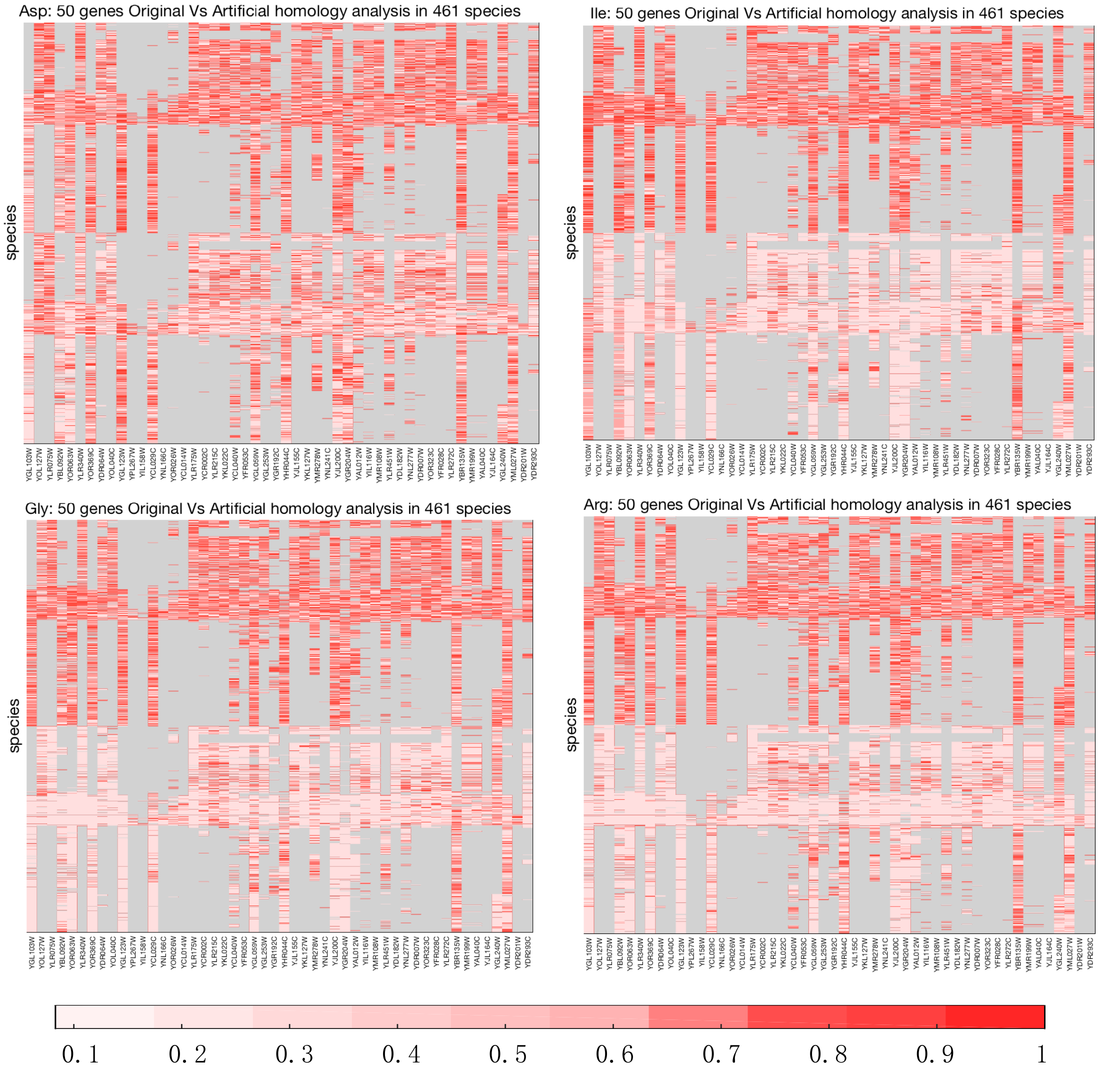


Figure 12: *Sn* based CUB measure of orthologs across 461 species corresponding to 50 genes in *S.cerevisiae* for 4 amino acids (Asp, Ile, Gly and Arg); along x axis are grouped 50 genes; along y axis 461 species containing the ortholog for the individual 50 genes. The 461 species are ordered according to phylogenetic tree of Fungi kingdom. The smaller *Sn* value is, The lighter its red shade shows; Grey means no such homology found or no such codon existed decoding such amino acid. Above half are patterns of real genomes, and the lower half are patterns of equal replaced artificial genomes.

# Chapter 4

# Genome Wide Codon Usage Bias Analysis

In the previous chapter, we introduced $Sn$ which quantifies the codon usage bias for each individual amino acid type in a gene. Next we wish to extend this to a global analysis of the genome. A feasible and valid method to combine all the $Sn$ values is required to perform such genome wide CUB analysis.

In section 2.3.3 of the literature review, we introduced various CUB measures, all of which first count synonymous codon occurrences in a predefined sequence context as the basic measure, then adopt different methods to combine these basic measures into one measure capable of quantifying CUB at the level of genome. Combination of all the $Sn$ values is obliged to consider the length differences and amino acid compositions.

In this chapter, we first propose the method to summarising $'Sn'$ into a genome wide CUB measure: $\mathcal{MD}$, which contains CUB information of each gene through the whole genome. Then we apply the measure $\mathcal{MD}$ to the real genomes of 462 fungal species. We find that there exists correlation between CUB and phylogenetic distances among species under certain scenarios.

## 4.1 Method for Summarising $Sn$ Within Genomes

One way in which $Sn$ can be characterised in a genome-wise manner is by quantifying the differences between observed $Sn$ and expected $Sn$ ($\overline{Sn}$) values. $\overline{Sn}$ is defined as the expected value of available $Sn$ values for a certain length (see section 2.1.6), therefore if the $\overline{Sn}$ is calculated based on all the possible $Sn$ values corresponding to a length, we denote it as theoretical $\overline{Sn}$. If the $\overline{Sn}$ is the mean value calculated based on the observed $Sn$ values in a genome, we denote it as empirical $\overline{Sn}$.

### 4.1.1 Attributes of $Sn$ Distributions Under Specific Assumptions

If the subsequence composition is subject to a systematic bias, then the sum of absolute differences between the empirical $\overline{Sn}$ values and the theoretical $\overline{Sn}$ values through the whole genome will depend on the magnitude of that bias. This is illustrated in Figure 13, which compares the distribution of $Sn$ values in the real genome to those in the two types artificial genomes (see section 2.2.4). We randomly chose one amino acid type from each synonymous codon family, and displayed the four types of amino acid as the representatives.

In Figure 13 it is apparent that the $Sn$ distribution of real genomes and that of artificial genomes without any bias is different, which confirms the view that the codon composition in real organisms is subject to systematic bias. Further to explore whether the difference between observed and expected $Sn$ in real genomes is due to the acknowledged global codon usage bias, we compared the real genome to the artificial genome in which the average global codon usage bias is the only acting force. Such comparison reveals that treating all the genes with the same global codon usage bias averages away variations of different CUB patterns of individual genes.

$Sn$ distribution is closely correlated to length, therefore to quantify genome wide CUB we include the information of subsequence length by plotting the theoretical $\overline{Sn}$ and the empirical $\overline{Sn}$ against their subsequence lengths separately. We define the curve depicting the relationship between subsequence lengths and the theoretical $\overline{Sn}$ as Q (see Figure 14), while define the scatter plot depicting the relationship between subsequence lengths and the empirical $\overline{Sn}$ as P. Q is derived from the theoretical calculation and is independent of the empirical CUB in the real genomes, and hence we adopt Q as the reference to evaluate different P derived from different genomes. As shown in Figure 15 the red P was derived from the real genome, the yellow P was derived from the artificial genome with

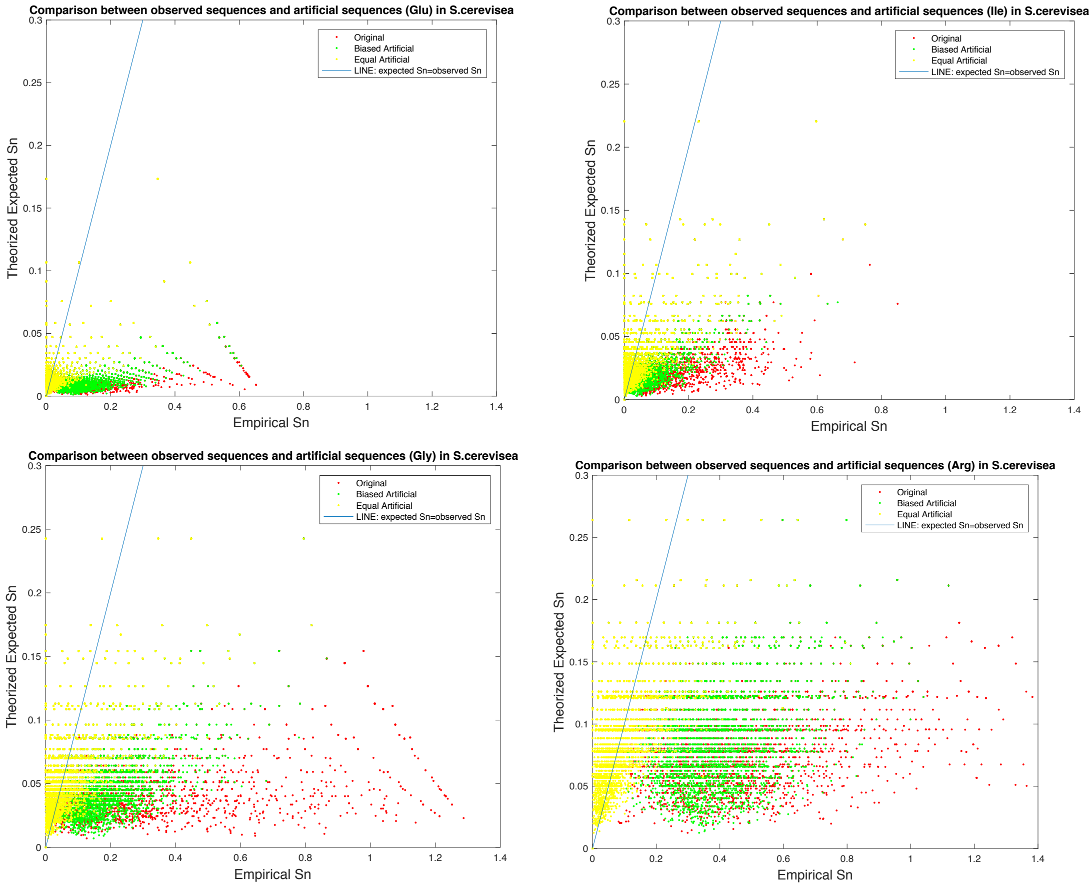


Figure 13: $Sn$ distribution overview: example of Sn values of all the subsequence encoding Glu, Ile, Gly and Arg in *S. cerevisiae*; along x axis are observed $Sn$ values, along y axis are corresponding theoretical $\overline{Sn}$ values, and the blue diagonal presents dots with the same $Sn$ value as corresponding $\overline{Sn}$. Comparison between observed and random generated artificial sequences: red dots represent values for the real genome, which are prone to take up the high $Sn$ value region and deviate from the blue diagonal; yellow dots represent values for artificial genome with random codon usage, which are spread symmetrically around blue diagonal in the low Sn value region; green dots represent values for artificial genome with random but weighted codon usage according to observed global codon usage frequencies, whose performance rank between yellow and red dots.

no codon usage bias, and the azure P was derived from the artificial genome with a unified global codon usage bias.

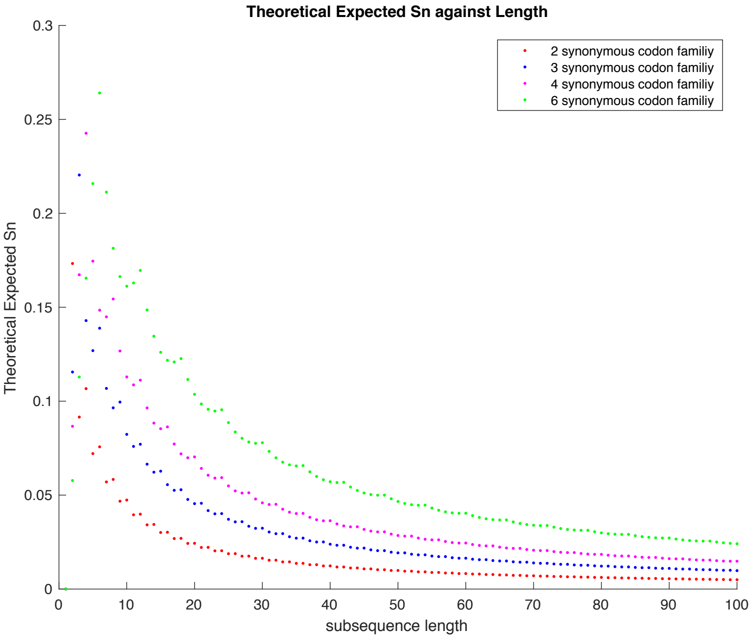


Figure 14: Reference curve Q: Relationship between theoretical $\overline{Sn}$ and its corresponding length for different synonymous codon families. X axis represents sequence length and Y axis represents theoretical $\overline{Sn}$. Curve Q takes into account exhaustive Sn values for $\overline{Sn}$ calculation at any subsequence length.

Figure 15 compares the bias strengths in the real *S. cerevisiae* genome among observed sequences and artificial sequences while adopting curve Q as the reference. The yellow P is the closest one to the reference Q. By contrast, the red P has the largest deviation from the reference Q. In addition the azure P ranks between the one for real genome and reference Q. Comparisons among different P reveal that in the real genome, there exists codon usage bias but such bias does not follow a unified pattern at the whole genome level.

### 4.1.2 $\overline{Sn}$ Based Genome Wide Measure

By comparing real genome and artificial genomes, we can see that summing absolute differences between theoretical and empirical $\overline{Sn}$ values at each individual length can be an option to present CUB within the genome. Next we specify a new genome wide measure of CUB as follows:

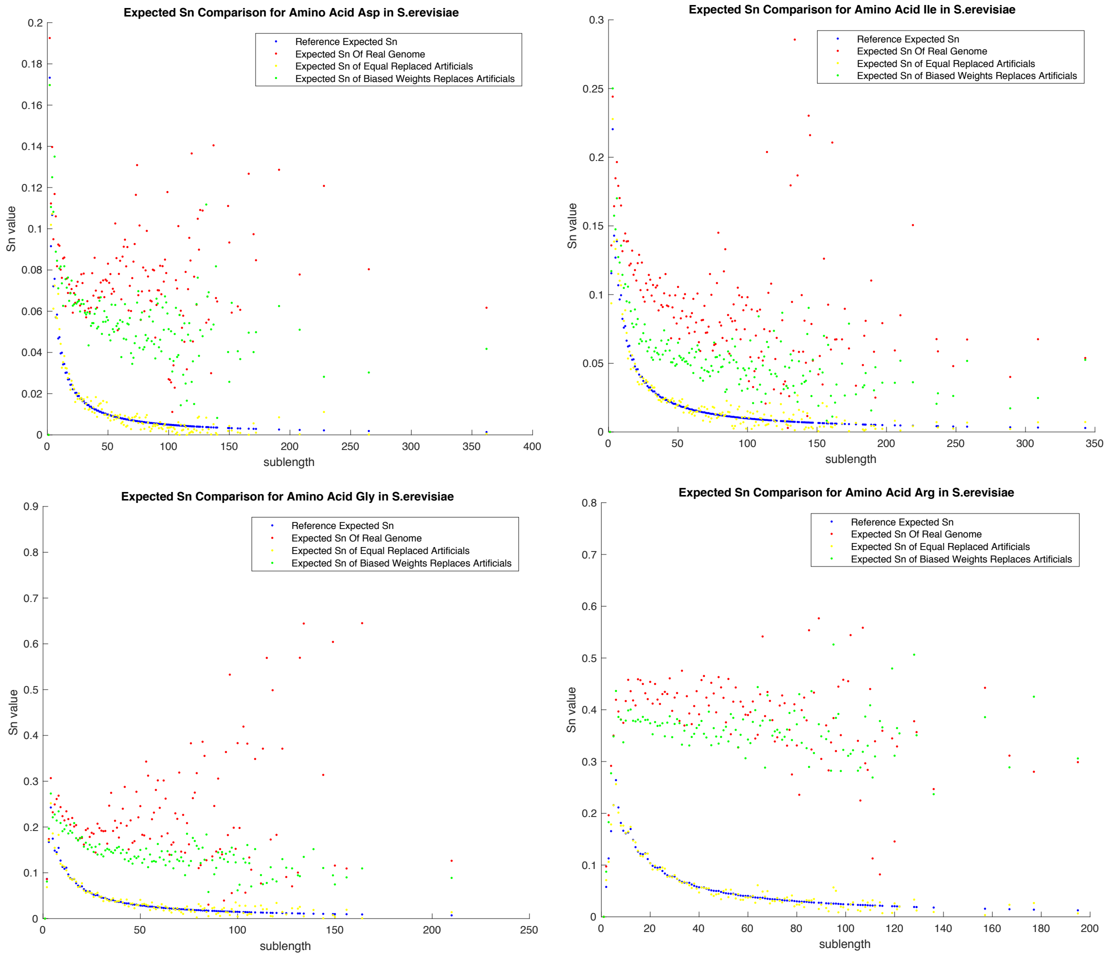


Figure 15: 4 example amino acids in *S. cerevisiae* genome. Blue dots represent the reference curve Q. Red dots represent the P in the real genome, which deviated most from curve Q; Yellow dots represent the P in the artificial genome with equal random codon usage, which spread systematically close to curve Q; Azure dots represent the P in the artificial genome whose codon usage probabilities are consistent with the observed global codon usage frequencies, which ranks between red and yellow P.

(1) Calculate the amino acid specific measure $MD$ at the whole genome level. Supposing the vector $\mathbf{P}$ is composed of the empirical $\overline{Sn}$ values for each subsequence length $L_i$ across the investigated genome, and the vector $\mathbf{Q}$ is composed of theoretical $\overline{Sn}$ corresponding to $L_i$. $L_L$ is the count of different $L_i$ within the genome. $MD$ reflects the distance between $\mathbf{P}$ and $\mathbf{Q}$.

The distance between $\mathbf{P}$ and $\mathbf{Q}$ can be calculated adopting different algorithms. The most widely accepted is the Minkowski distance as follows:

$$D(\mathbf{P}, \mathbf{Q}) = (\sum_{i=1}^{n} (|\mathbf{P}_i - \mathbf{Q}_i|)^p)^{1/p} \tag{5}$$

where the Euclidean distance ($p = 2$) and Manhattan distance ($p = 1$) are specific instances.

Manhattan distance is less sensitive to one dimension of an extremely high difference between $\mathbf{P}$ and $\mathbf{Q}$ (Aggarwal, Hinneburg and Keim (2001)). In our study, each dimension of $\mathbf{P}$ and $\mathbf{Q}$ contain useful information about CUB for each length, and hence our measure prefers to maintain as many attributes of dimensions as possible rather than one particular attribute of that dimension. Therefore we chose Manhattan distance (p=1) for $MD$ calculation and defined the $MD$ value as the Manhattan distance between $\mathbf{P}$ and $\mathbf{Q}$ with further normalisation by $L_L$:

$$MD := \left( \sum_{i=1}^{L_L} |\mathbf{P}_i - \mathbf{Q}_i| \right) / L_L \tag{6}$$

where $\mathbf{P}_i$ is the empirical $\overline{Sn}$ for the length $L_i$ and $\mathbf{Q}_i$ is the theoretical $\overline{Sn}$ for the length $L_i$.

The measure $MD$ makes CUB comparable in genomes with varied gene lengths. Further normalisation by $L_L$ diminishes the impact from the gene length diversities.

(2) We further define $\mathcal{MD}$, a genome wide CUB measure as $\mathcal{MD} := [MD_E$, $MD_H$, $MD_Q$, $MD_F$, $MD_Y$, $MD_C$, $MD_N$, $MD_K$, $MD_D$, $MD_I$, $MD_P$, $MD_T$, $MD_A$, $MD_V$, $MD_G$, $MD_L$, $MD_S$, $MD_R]$. The subscripts of $MD$ are the amino acid abbreviations. $\mathcal{MD}$ contains CUB information for all the 18 amino acids among all genes through a genome.

Hitherto we have proposed the sequence specific CUB measure $Sn$, amino acid specific CUB measure $MD$, and genome wide CUB measure $\mathcal{MD}$. In chapter 2 we have demonstrated the application of $Sn$ for sequence specific CUB analysis, next we introduce the application of $\mathcal{MD}$ for CUB analysis in the genome-wise manner.

### 4.1.3 Results of $\mathcal{MD}$ Application

We applied $\mathcal{MD}$ measure to the *S. cerevisiae* genome and obtained the results in Figure 16. $MD$ values for the 18 amino acids with more than one codon choice in real *S. cerevisiae* genomes are higher than the artificial genome generated by random codon replacement, which confirms that synonymous codons are not equally randomly spread in real genome. Furthermore, the real *S. cerevisiae* genome also has higher $MD$ values compared to the artificial genome where codons are replaced randomly with weights corresponding to the global codon usage bias. A potential explanation for this effect is that codon usage bias drive individual sequences in different directions. For the majority of sequences, codons that are preferred on a genome-wide scale are also preferred for the individual sequence. However, for a minority of sequences, codons may be preferred that are non-preferred on a genome-wide scale (Neafsey and Galagan (2007)). Because of the way global codon usage bias is calculated, these opposing effects would be averaged away. In contrast, $\mathcal{MD}$ method treats each gene individually based on $Sn$, which avoids averaging away CUB driving forces acting on genes in different directions.

After demonstrating how the $\mathcal{MD}$ method quantifies CUB across a genome, we apply it to 462 species in the Fungi kingdom. A heatmap obtained as Figure 17 displays bias magnitude of each species and shows different trends for different amino acids within one genome and also the same amino acid among different genomes. There is a general trend for higher $MD$ values with higher synonymous codon choices, for example, average $MD$ value is higher for the amino acids encoded by six different synonymous codons compared to amino acids encoded by only two different synonymous codons. There is also variation between amino acids encoded by the same number of codons. For example, cysteine (C) has lower $MD$ values than other amino acids encoded by 2 codons, whereas leucine(L) show high MD values compared to the other six-codon amino acids, implying that its usage is very non-random.

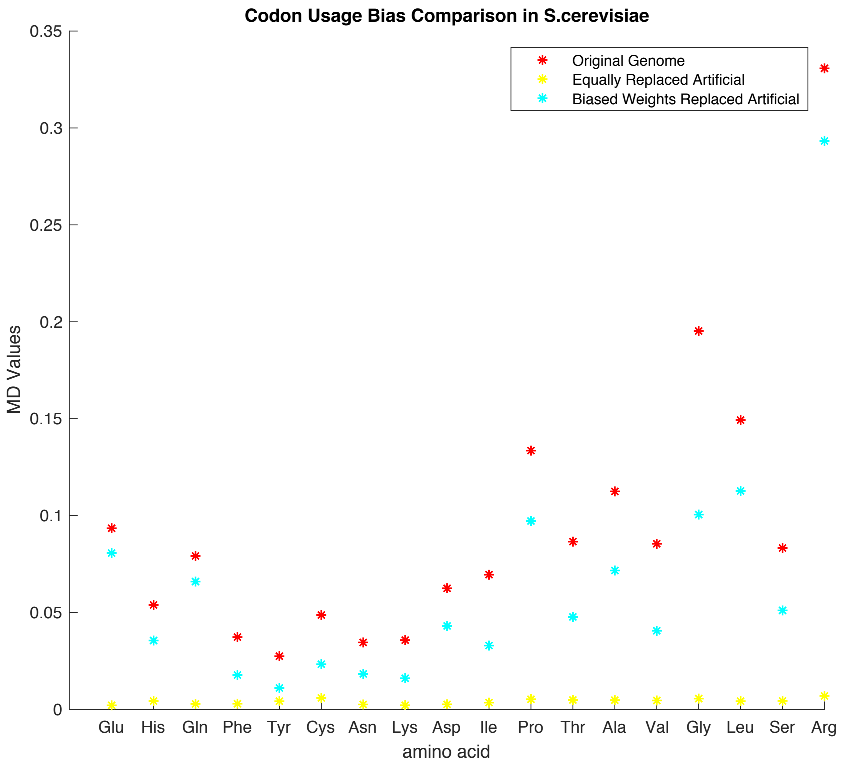


Figure 16: $MD$ values for 18 amino acids in species *S. cerevisiae*. $MD$ value for each amino acids, is summarised from differences between vector P and vector Q, articulates codon usage bias strength for each amino acid in the genome. $MD$ values of red dots for real genome are higher than azure dots which represents artificial genome with a unified global codon usage. At the lowest level the yellow dots represent the artificial genome with equal synonymous codon usage.

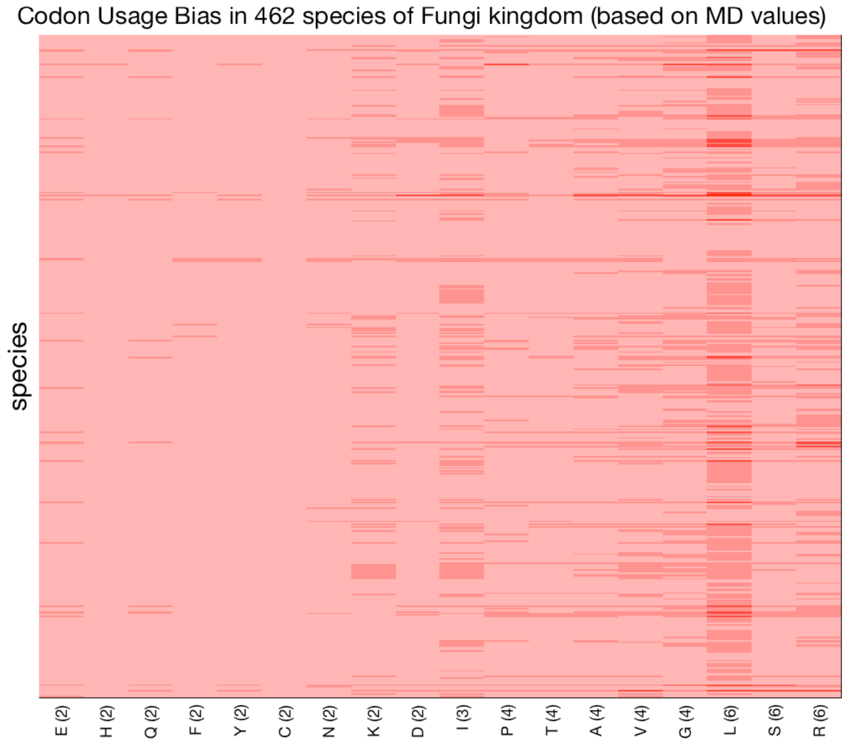


Figure 17: $MD$ values for 18 amino acids of 462 species among Fungi kingdom. Sizes of synonymous codon families are labelled in the parentheses behind each amino acid abbreviations. Y axis corresponds to the analysed 462 species ordered according to Phylogenetic tree. The darker the shade of red, the higher the $MD$ value.

## 4.2 Self-Organising Map for Genome Wide CUB Analysis Based on $\mathcal{MD}$

The heatmap in Figure 17 based on $\mathcal{MD}$ values shows amino acid specific and species specific patterns of codon usage bias. We now define the genome wide CUB measure for a species 'sp' as: $\mathcal{MD}_{sp} = [MD_E^{sp}, MD_H^{sp}, MD_Q^{sp}, MD_F^{sp}, MD_Y^{sp}, MD_C^{sp}, MD_N^{sp}, MD_K^{sp}, MD_D^{sp}, MD_I^{sp}, MD_P^{sp}, MD_T^{sp}, MD_A^{sp}, MD_V^{sp}, MD_G^{sp}, MD_L^{sp}, MD_S^{sp}, MD_R^{sp}]$.

If the CUB measure considers all the 18 amino acids as one whole attribute, previous work used the 'weighted sum' to combine each amino acid specific measure into a single value as a measure of the genome wide CUB (Suzuki, Saito and Tomita (2004), Urrutia and Hurst (2001)), where weights are chosen considering the amino acid composition in the genome. However the simple linear combination lacks a concrete mathematical validation and may lose amino acid specific information for the genome wide CUB analysis.

In the following two sections, we apply two unsupervised machine learning technics 'Self Organising Map' and 'Hierarchical Cluster' to study CUB driven factors adopting the high dimensional variable $\mathcal{MD}$.

First we explain how to use 'Self Organising Map' adopting $\mathcal{MD}$ to analyse CUB pattern across species.

### 4.2.1 Self Organising Map Approach

If we have a high dimensional input feature space where the input variables live, and we have little knowledge of what correlation to expect, then it is extremely difficult to identify the feature relationships among the input variables. Our CUB datasets of $\mathcal{MD}$ among species constitute such high dimensional CUB feature space with unknown relationships between inputs. To explore such input feature space Self Organising Maps (SOM for short) is a feasible way.

SOM borrows ideas from biological models of neural systems and are widely accepted as an unsupervised learning technique based on competitive learning algorithms. The aim of SOM is to map from a high dimensional input space to a two dimensional output space, where the topology of the input space is reflected and visualised in the two dimensional output space. Such transformation makes it possible to explore the relationships among high dimensional variables. To be specific in our case SOM provides a way to visualize the original high dimensional CUB features on a two dimensional plane based on which it is easy to make further calculations to compare input CUB information of species. The process of this transformation is called training process based on the input variables. The result of SOM can be thought of the combination of the dimensionality deduction and clustering of the inputs.

SOM projects high-dimensional data information onto two-dimensional flat maps which are composed of nodes. All the nodes are arranged in a regular shaped grid like rectangle or hexagon, and each node is defined by a weight vector which represents this node's position in the input space and can be used to calculate distance to input variables shown in Figure 18(a). The purpose of such projection is to form the two-dimensional map with nodes who maintain the original input topology which process is illustrated in Figure 18(b).

The workflow of SOMs to form the two dimensional map is generalised as follows:

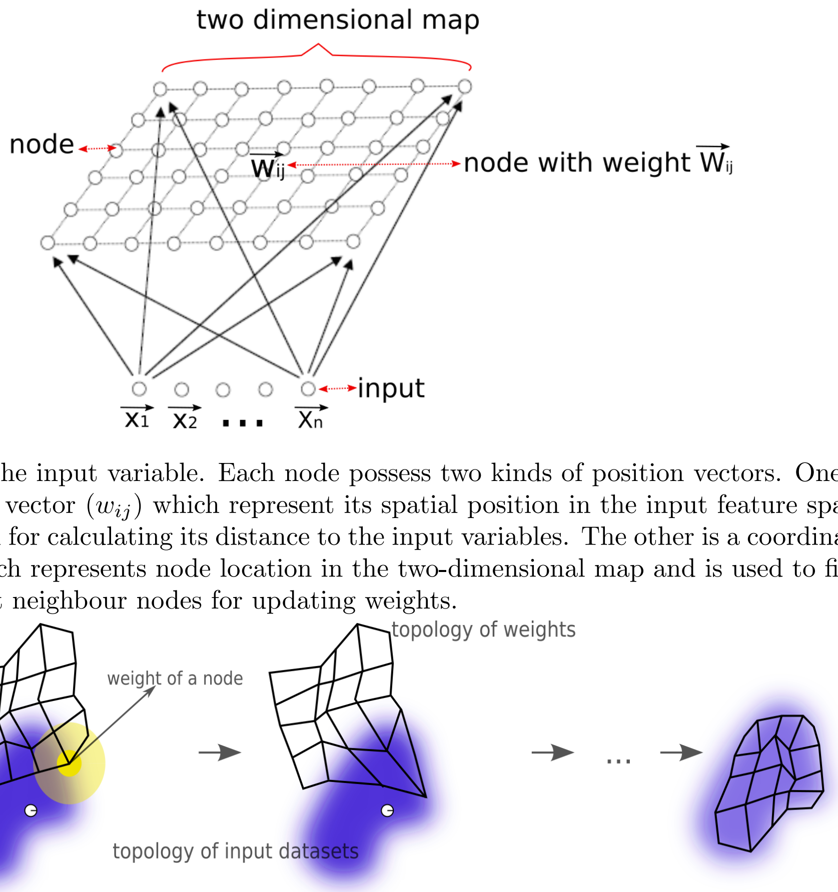
Step 1: Initialisation. We choose to initialise the nodes with weight vectors of random small values (Akinduko and Mirkes (2012)).

Step 2: Competition. This step consists of a number of sub-steps. (1) Choose one input variable, calculate its distances to all the nodes on the two dimensional map, and then find the nearest node and its predefined neighbours as the winning nodes to be updated. (2) The selected nodes are updated by adjusting their weights and hence moving towards the input variable according to equation:

$$w_i^{new} = w_i^{old} + \alpha_{ik}\Delta w_i^{old} = w_i^{old} + \alpha_{ik}\theta(x - w_i^{old}), i = 1, 2, ..., m$$

$$\alpha_{ik} = \begin{cases} 1, & d_ik < T. \\ 0, & otherwise. \end{cases} \tag{7}$$

where $\alpha_{ik}$ is the neighbourhood function which defines only neighbours of distance less than T are active neighbours to be altered weights, $w_i$ is the node weight, x is the input variable, $\theta$ is the learning rate. The neighbourhood distance calculation adopts the nodes position coordinate on the two dimensional map rather than nodes weight vectors. T and $\theta$ gradually reduce at a predefined rate, because when the nodes are adjacent to the input variables, the finer nudge of nodes'


two dimensional map
node
$\vec{W}_{ij}$
node with weight $\vec{W}_{ij}$
input
$\vec{X}_1$ $\vec{X}_2$ ... $\vec{X}_n$
weight of a node
topology of weights
topology of input datasets
...

(a) $\vec{X}_n$ is the input variable. Each node possess two kinds of position vectors. One is the weight vector ($w_{ij}$) which represent its spatial position in the input feature space and is used for calculating its distance to the input variables. The other is a coordinate vector which represents node location in the two-dimensional map and is used to find the nearest neighbour nodes for updating weights.

(b) The nodes on the two dimensional map have their corresponding spatial topology (the mask) in the same space as input topology (the purple area). When nodes are trained, in each iteration the selected node (yellow spot) is moved towards to the input variable (white spot) by adjusting its weight. The final ideal result is the mask resembles the purple area.

Figure 18: Illustration of how SOM works. Through training, topology of the input space are reflected by the two dimensional map. We then classify input variables according to their geographical distance to nodes on the map.

weights is required with shrinking learning rates and neighbourhoods.

Step 3: Termination. Repeat step 2 for a pre-defined number of iterations. The iteration is selected as 500 times the number of nodes, because two dimensional map is fine tuned and comes to provide an accurate statistical quantification of the input space when the iterations reach this value (Polani (2002)). Results are illustrated on the 'sample plane' and 'neighbourhood distances'. Sample plane displays the distances between input variables to all the nodes on the two dimensional map, by which we can compare the feature similarity between input variables. The neighbourhood distances display the node geometry in the feature space trained by the input variables, by which we can cluster the input variables and visualise the input space topology.

### 4.2.2 Results of Self Organising Map Application

For SOM analysis, the input variables are species, and each input variable has $\mathcal{MD}$ as its 18 dimensional features [$MD_E^{sp}$, $MD_H^{sp}$, $MD_Q^{sp}$, $MD_F^{sp}$, $MD_Y^{sp}$, $MD_C^{sp}$, $MD_N^{sp}$, $MD_K^{sp}$, $MD_D^{sp}$, $MD_I^{sp}$, $MD_P^{sp}$, $MD_T^{sp}$, $MD_A^{sp}$, $MD_V^{sp}$, $MD_G^{sp}$, $MD_L^{sp}$, $MD_S^{sp}$, $MD_R^{sp}$]. We selected 20 species listed in Table 15 as input variables.

We selected these 20 species because they form two phylogenetic groups where members of each group are closely related in the phylogenetic tree, but are not closely related between groups (see the phylogenetic tree in Figure 19). The pattern of the species relationship makes it possible to evaluate how CUB changes over both short and long distances in evolution. To assist visual inspection, we generated the corresponding phylogenetic tree based on the phylogenetic distance of these 20 species. The phylogenetic tree is created using online tool [1].

Figure 20(a) illustrates the distances between input variables to the nodes on the two dimensional map. Imagine that two input variables have very similar features, then they tend to position in a very close location in the input feature space, equivalently the pattern of distances between nodes to these two variables tend to be similar. Therefore patterns evident on the sample plane reveal how similar the input variables are based on the considered features. To be specific, our input variables are 20 species and each species has the feature of CUB represented by $\mathcal{MD}$, and the sample plane indicates the similarity between species

[1] https://phylot.biobyte.de/

Table 15: 20 species for SOM analysis

| Group Name | Species Name | Abbreviation |
|---|---|---|
| Saccharomycetales | Saccharomyces arboricola_$h$_6 | S.arboricola |
| | Saccharomyces eubayanus | S.eubayanus |
| | Saccharomyces kudriavzevii | S.kudriavzevii |
| | Ashbya gossypii | A.gossypii |
| | Yarrowia lipolytica | Y.lipolytica |
| Schizosaccharomycetales | Schizosaccharomyces pombe | S.pombe |
| | Schizosaccharomyces japonicus | S.japonicus |
| | Schizosaccharomyces octosporus | S.octosporus |
| Eurotiales | Aspergillus clavatus | A.clavatus |
| | Aspergillus flavus | A.flavus |
| | Aspergillus nidulans | A.nidulans |
| | Aspergillus niger | A.niger |
| | Aspergillus oryzae | A.orizae |
| | Aspergillus terreus | A.terreus |
| | Aspergillus fumigatus | A.fumigatus |
| Hypocreales | Fusarium fujikuroi | F.fujikuroi |
| | Fusarium graminearum | F.graminearum |
| | Fusarium oxysporum | F.oxysporum |
| | Fusarium verticilloides | F.verticilloides |
| | Trichoderma reesei | T.reesei |

based on the feature of CUB. From Figure 20(a) CUB patterns clearly reproduce some features of the phylogenetic relationship between species, for example, the three *Saccharomyces* species display a minimum value (bright yellow hexagon) in the bottom right hand corner of the plot, whereas the other two species in the *Saccharomycetales* order which are more distantly related to the three *saccharomyces species* display a more centrally located minimum. Interestingly however, this pattern does not always hold. For example, the three *Schizosaccharomyces* species display stronger variation in the SOM patterns.

Figure 20(b) display the spatial distances between nodes in the feature space which resemble the input variables. If the nodes on the two dimensional map highly resemble the input feature space, the clustering property displayed by these nodes should convey the clustering property of the input feature space. Because nodes are arranged in a two dimensional plane, the pattern of distances between nodes are more convenient for visualisation. To be specific in our case, from the

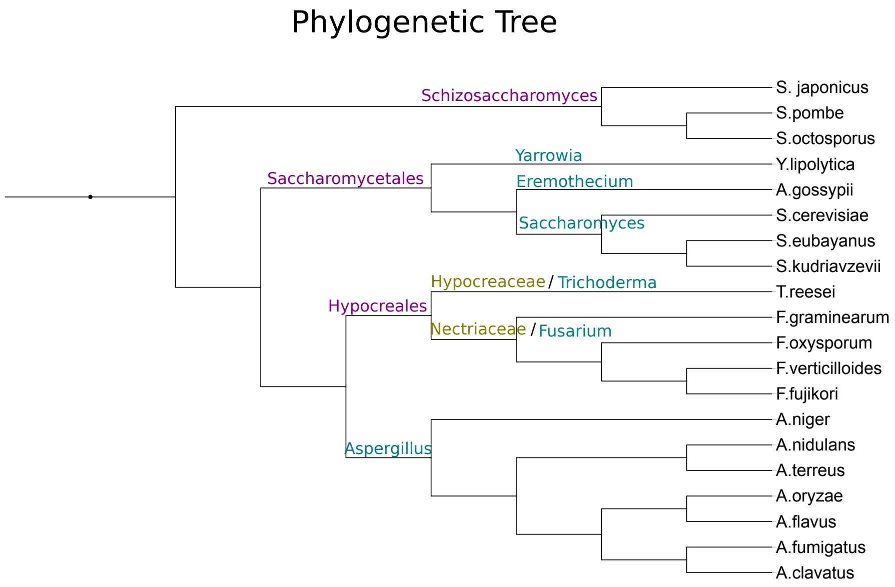


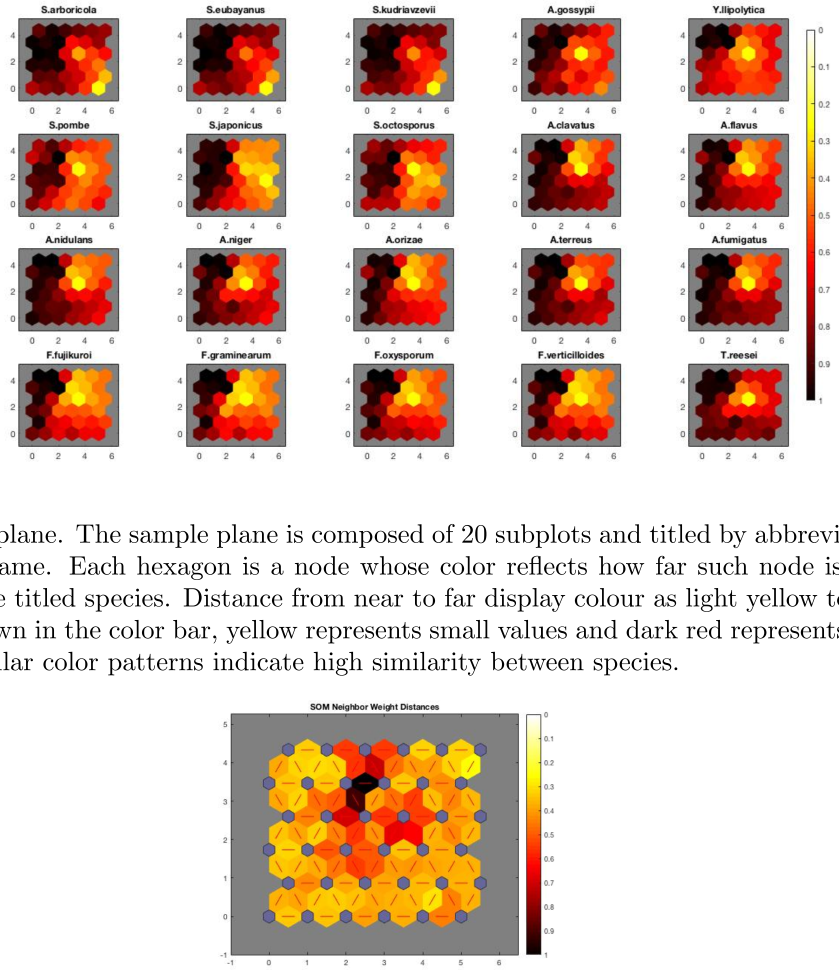
Figure 19: Phylogenetic tree of 20 species

two dimensional map we can see 2 distinct regions which convey that based on the CUB pattern the input 20 species should be divided into two groups.

SOM makes it possible to consider $\mathcal{MD}$ as a whole attribute of individual input species, without combining different $MD$ values into a single number.

Now consider if the input variable is any entity which possess the property of $MD$, SOM results are able to reveal CUB patterns among the input variables. To be specific, consider the 18 amino acids as input variables, and each amino acid has 20 features formed by 20 species. For example for amino acid E, its 20 features are displayed as $[MD_E^{sp_1}, MD_E^{sp_2}, MD_E^{sp_3}, ..., MD_E^{sp_i}, ..., MD_E^{sp_{19}}, MD_E^{sp_{20}}]$ $(i \in [1, 20])$. In this case the SOM results are able to show amino acid specific patterns among 20 species, and further reveal whether amino acids coevolve with each other among different species.

As shown in Figure 21(a), the sample plane conveys the similarity among input variables here as 18 amino acids. A number of amino acids show similar patterns (Glu, Gln, Tyr, Asn, Pro, Thr, Ala, and Ser) other amino acids differ strongly from this main pattern. Some of these patterns seem unique to individual amino acids (His, Phe, Cys). Interestingly, the chemically related branched-chain amino acids Leu, Val, and Ile show patterns that are more similar between these amino acids than to other amino acids. This indicates that the different CUB patterns may reflect chemical relationship between amino acids.



(a) Sample plane. The sample plane is composed of 20 subplots and titled by abbreviations of species name. Each hexagon is a node whose color reflects how far such node is away from the the titled species. Distance from near to far display colour as light yellow to dark red. As shown in the color bar, yellow represents small values and dark red represents large values. Similar color patterns indicate high similarity between species.

(b) Node neighborhood distance. In this figure the blue hexagons are nodes on the two dimensional map. The red lines indicate the distance between nodes whose magnitude is displayed by the colors around the red lines. Neighbour distance shows distance between nodes in the feature space, where distances from near to far display colour as light yellow to dark red. As shown in the color bar, yellow represents small values and dark red represents large values. Nodes neighbour distance conveys information about how many clusters input species may divide into, and here we can see 2 distinct codon usage bias pattern groups are divided by the dark red curve which vertically passes through the middle of the map.

Figure 20: 20 species as input variables, each input variable has 18 dimensions (18 MD values corresponding to 18 amino acids).

Nodes neighborhood distance shown in Figure 21(b) conveys the clustering property of the input variables based on the investigated features, to be specific in our case, Figure 21(b) suggests how the 18 amino acids can be clustered based on amino acid specific CUB features from 20 species. There is a distinct cluster divided by dark red curves in the left middle part of the map, and also a distinct cluster in the right upper corner of the map.

To sum up, by selecting input variable and variable features, SOM is able to reveal codon usage bias pattern for the specific purposes.

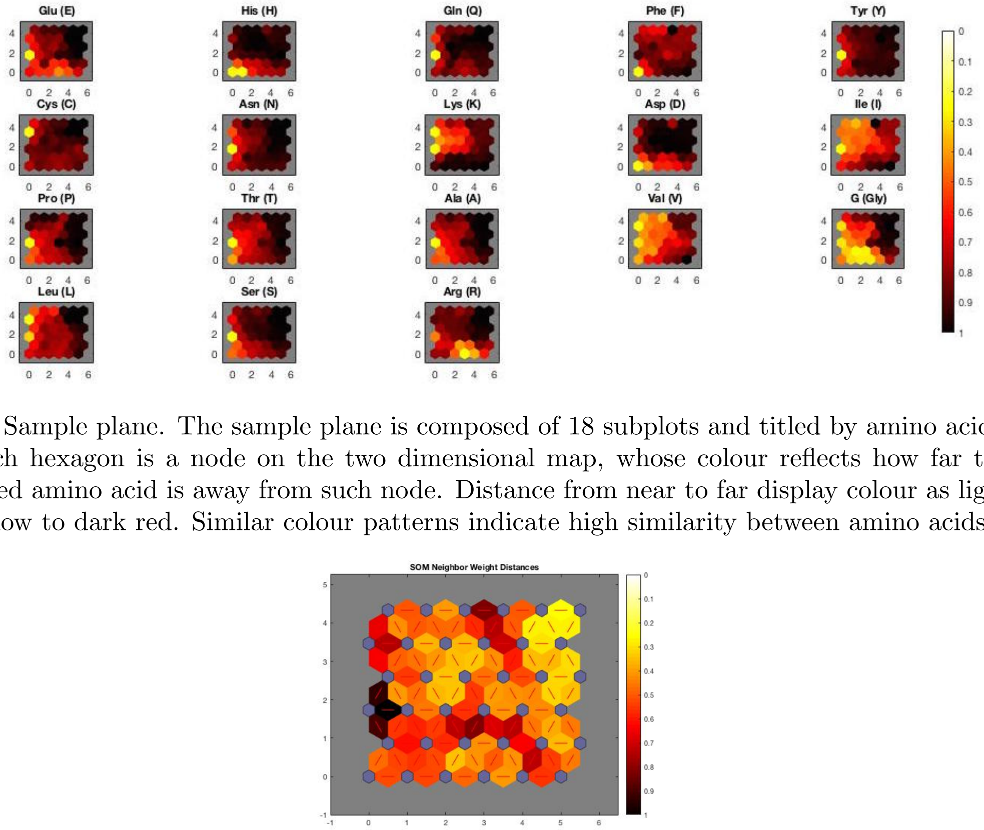
## 4.3 Hierarchical Clustering for Genome Wide CUB Analysis Based on $\mathcal{MD}$

Next we introduce the concept of 'hierarchical cluster' to perform CUB analysis, which will reveal patterns of underlying relationships not directly visible in numerical data. When the hierarchical cluster tree is constructed based on some features of a set of species, such cluster tree will represent relationships among these species based on the input features. We built the hierarchical cluster tree based on $\mathcal{MD}$ values and taxonomy of species, and then the cluster tree derived from $\mathcal{MD}$ values represent the CUB relationships among species and the cluster tree derived from the taxonomy represent phylogenetic relationships among species. By comparing similarity between these two types of cluster trees, we found that CUB prone to correlate to phylogenetic distances for remote groups of species, equivalently phylogenetic distance correlates to CUB from the perspective of a wide evolutional span.

Now we will introduce how to construct the hierarchical trees and compare their similarity in the next two sections.

### 4.3.1 Hierarchical Clustering Approach

The aim of Hierarchical Clustering is to create a dendrogram with multilevel hierarchy (Maimon and Rokach (2009)). The basic idea of Hierarchical Clustering is to sequentially pair and join nodes with the closest proximity to form a new cluster, which serves as a new node to participate in the following merging procedure until all the nodes are involved to create the hierarchical cluster tree.



(a) Sample plane. The sample plane is composed of 18 subplots and titled by amino acids. Each hexagon is a node on the two dimensional map, whose colour reflects how far the titled amino acid is away from such node. Distance from near to far display colour as light yellow to dark red. Similar colour patterns indicate high similarity between amino acids.

(b) Node neighbourhood distance. The blue hexagons represent nodes on the two dimensional map. The red lines within a color region represents distances between connected nodes in the feature space. Distances from near to far display colour as light yellow to dark red. Nodes neighbour distance conveys how many clusters the input amino acids may divide into.

Figure 21: 18 amino acids as input variables, each has 20 dimensions (20 amino acid specific MD values corresponding to 20 species).

The Hierarchical Clustering workflow can be summarised as follows:

Step 1: Distance calculation. Compare the similarity between each pair of input nodes by calculating the Euclidean distance between them. Based on all the input nodes, we obtained the original distance matrix whose entries are the distances between paired nodes.

Step 2: Forming a new cluster. Initially consider each input as a single cluster. In each iteration, a pair of clusters which have the closest distance are joined together to form a new cluster. The new formed cluster participates in the next merging procedure.

There are many algorithms to calculate the distance between two clusters p and q. Here we determine the cluster distance by choosing the smallest distance between the nodes separated in the two clusters:

$$d(p,q) = \min(\mathtt{dist}(\mathtt{x_{pi}}, \mathtt{x_{qj}})), \mathtt{i} \in \mathtt{p}, \mathtt{j} \in \mathtt{q} \tag{8}$$

here $p$ and $q$ are two clusters, i is the $i$-th node in cluster p and j is the $j$-th node in cluster q.

Each iteration distance matrix updates distances between newly formed cluster and other existing clusters. Steps 1 and 2 are repeated until a hierarchical tree is formed.

Step 3: Pruning the hierarchical tree. Merge nodes below a predefined threshold into one single cluster. The threshold can be the nodes height, or maximum cluster numbers and ect. We use maximum cluster numbers as the threshold to prune hierarchical trees.

### 4.3.2 Cluster Similarity Quantification

Here we use an approach to quantify the similarity between cluster trees based on the algorithm recommended by E.B.Fowlkes in 1983(Fowlkes and Mallows (1983)). The workflow is as follows:

Step 1: Obtain two hierarchical cluster trees based on different features of input variables.

Step 2: Cut the two cluster trees individually so that they have the same cluster numbers n. Count the same input variable amounts within each paired clusters and form the matching matrix $\mathbf{M}$ as follows:

$$\mathbf{M} = [m_{ij}](i = 1, ..., k; j = 1, ..., k; k = 2, ..., n - 1) \tag{9}$$

where the element $m_{ij}$ represents the number of common entries (namely the same input variables) between the $i$-th cluster in the first cluster tree and the $j$-th cluster in the second cluster tree.

Step 3: Using $\mathbf{M}$ to calculate the similarity score $B_k$ which represents the similarity between clusters:

$$\begin{aligned}
B_k &= \frac{T_k}{\sqrt{(P_k)(Q_k)}} \\
T_k &= \sum_{i=1}^{k}\sum_{j=1}^{k} m_{ij}^2 - n \\
P_k &= \sum_{i=1}^{k} m_{i.}^2 - n \\
Q_k &= \sum_{j=1}^{k} m_{.j}^2 - n \\
m_{i.} &= \sum_{j=1}^{k} m_{ij} \\
m_{.j} &= \sum_{i=1}^{k} m_{ij}
\end{aligned} \tag{10}$$

Step 4: Calculate the confidence interval under the null hypothesis of independent clusterings. Fix cluster tree structures and label entry node according to bivariate normal distributed random values, under which setting the two cluster trees should have no similarity. Based on randomly generated sample trees, we calculate the expected value $\mathrm{E}(B_k)$ and variance $\mathtt{var}(\mathtt{B_k})$ according to Equation

11:

$$
\begin{aligned}
E(B_k) &= \sqrt{P_k Q_k} n(n-1) \\
\mathtt{var(B_k)} &= \frac{\mathtt{2}}{\mathtt{n(n-1)}} + \frac{\mathtt{4P'_k Q'_k}}{\mathtt{n(n-1)(n-2)P_k Q_k}} + \\
&\frac{(P_k - 2 - 4P'_k/P_k)(Q_k - 2 - 4Q'_k/Q_k)}{n(n-1)(n-2)(n-3)} - \\
&\frac{P_k Q_k}{n^2(n-1)^2} \qquad (11) \\
P'_k &= \sum_{i=1}^{k} m_{i.}(m_{i.}-1)(m_{i.}-2) \\
Q'_k &= \sum_{j=1}^{k} m_{.j}(m_{.j}-1)(m_{.j}-2)
\end{aligned}
$$

Based on E($B_k$) and $\mathtt{var(B_k)}$, we deduce the confidence interval of $B_k$ (confidence level $\alpha$=0.05) for two independent trees: $E(B_k) \pm 2\sqrt{\mathtt{var(B_k)}}$.

Step 5: Based on the obtained $B_k$ value which suggests the similarity strength of the investigated trees, and also the confidence interval $E(B_k) \pm 2\sqrt{\mathtt{var(B_k)}}$ where the $B_k$ values suggest no correlation. We make the judgement whether we can accept there are correlation between investigated trees, in addition we measure the similarity according to the magnitude of $B_k$.

To better explain the meaning of the matching matrix **M** and cluster similarity score $B_k$, we give an example in Figure 22.

Matlab code to implement the cluster comparison is shown below:

```
function [ClusterSam,BoundUpLow] = clustering_comparison(ZZ1, ZZ2,kClust,spCount)

%%ZZ1,ZZ2 are the dendrogram matrices for individual cluster derived from MATLAB clustering package

ClusterSn1=cluster(ZZ1,'maxclust',kClust);
ClusterSn2=cluster(ZZ2,'maxclust',kClust);
%% ClusterSn: cluster label for each species in individual cluster

Cbase=(1:spCount); %% spCount: how many species
```

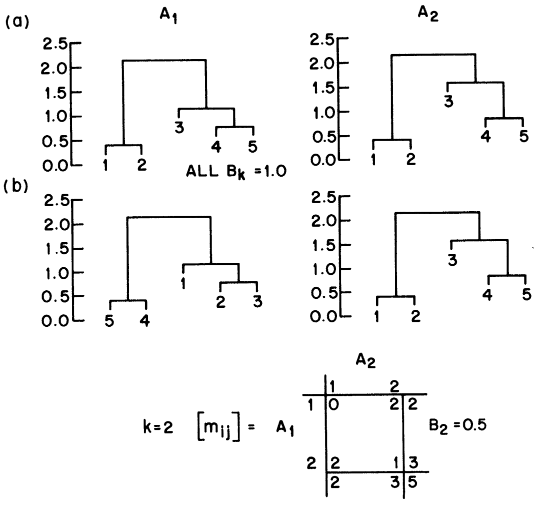


Figure 22: Dendrograms of two hierarchical cluster trees and the formation of matching $m_{ij}$ matrix for k=2. Figure (a) shows two identical hierarchical cluster trees and $B_k = 1$. In Figure (b), cut two clusters at the level of branches of 2. Compare the first two branches, there is no same element, therefore $m_{11} = 0$. Compare the first branch of the tree on the left and the second branch of the tree on the right, there are two same elements '4' and '5', therefore the entry in the first row and the second column of the matching matrix $m_{12}$ is 2. According to Equation 10, we calculate $B_k$=0.25. (Figure resource: Fowlkes and Mallows (1983))

```

for kClustCount=1:kClust
GrpClust1{kClustCount}=Cbase(ClusterSn1==kClustCount);
GrpClust2{kClustCount}=Cbase(ClusterSn2==kClustCount);
end

M=zeros(kClust,kClust); %% kClust decides where to prune the trees
for M1count=1:kClust
    for M2count=1:kClust
        M(M1count,M2count)=length(find((ismember(GrpClust1{M1count},GrpClust2{M2count}))));
%%evluate matrix: entry contains counts of common elements in two branches in GrpClust1 and GrpClust2
    end
end

Mi=sum(M,2);
Mj=sum(M,1);
n=sum(M(:));

T=sum(M(:).^2)-n;
P=sum(Mi(:).^2)-n;
Q=sum(Mj(:).^2)-n;
ClusterSam= T/sqrt(P*Q);
PP=sum(Mi(:).*(Mi(:)-1).*(Mi(:)-2));
QQ=sum(Mj(:).*(Mj(:)-1).*(Mj(:)-2));

meanB=sqrt(P*Q)/(n*(n-1));
%% the mean and std: random unrelated cluster trees.
varB=2/(n*(n-1))+4*PP*QQ/(n*(n-1)*(n-2)*P*Q)+(P-2-4*PP/P)*(Q-2-4*QQ/Q)/(n*(n-1)*(n-2)*(n-3))-P*Q/(n.^2*(n-1).^2);

BoundUpLow=[meanB-2*sqrt(varB),meanB+2*sqrt(varB)];
%% when Bk locate outside this area, means similaries is significance
```

```

end
```

It has been argued that the E.B.Fowlkes method takes into account topologies of individual trees, but does not consider the heights of internal nodes within individual tree (Wagner and Wagner (2007)). However our cluster trees are formed based on different types of data sets, therefore internal cluster heights cannot be meaningfully compared. In this sense putting aside the heights of internal nodes within the tree is reasonable and this quantification method satisfies our purpose well.

### 4.3.3 Results of Hierarchical Clustering Application

After introducing the algorithms to perform the correlation study between CUB and phylogenetic distance, we apply the algorithms to real genome data from Fungi kingdom when adopting $\mathcal{MD}$ as the genome wide CUB measure and taxonomy as the measure of phylogenetic distance. First we generate hierarchical clusters based on $\mathcal{MD}$ and species taxonomy separately (taxonomy information are retrieved from the NCBI taxonomy online tool [2]); secondly we quantify the similarity between the CUB cluster tree and phylogenetic cluster tree. If the similarity between these two trees is high, then this means that CUB has a high correlation with the phylogenetic distance among the investigated species set.

#### 4.3.3.1 Hierarchical Cluster Trees Based on $\mathcal{MD}$ and Species Taxonomy

Beginning with a small set of species, we chose 10 species in Table 15 as the input variables, where *Saccharomyces arboricola* (sahib), *Saccharomyces eubayanus* (sea), *Saccharomyces cerevisiae* (sc), *Saccharomyces kudriavzevii* (saku), *Saccharomyces sp* (sp) are phylogenetic close species in the *Saccharomycetales* order, and *Fusarium fujikuroi* (ff), *Fusarium oxysporum* (of), *Fusarium verticillioides* (fv), *Fusarium graminearum* (fg), *Fusarium poae* (fp) are phylogenetic close species in the *Hypocreales* order (abbreviations of species are displayed in the parenthesis after each species name). These 10 species are in 2 distant phylogenetic groups, and species within each group are close to each other.

[2] https://www.ncbi.nlm.nih.gov/Taxonomy/TaxIdentifier/tax_identifier.cgi

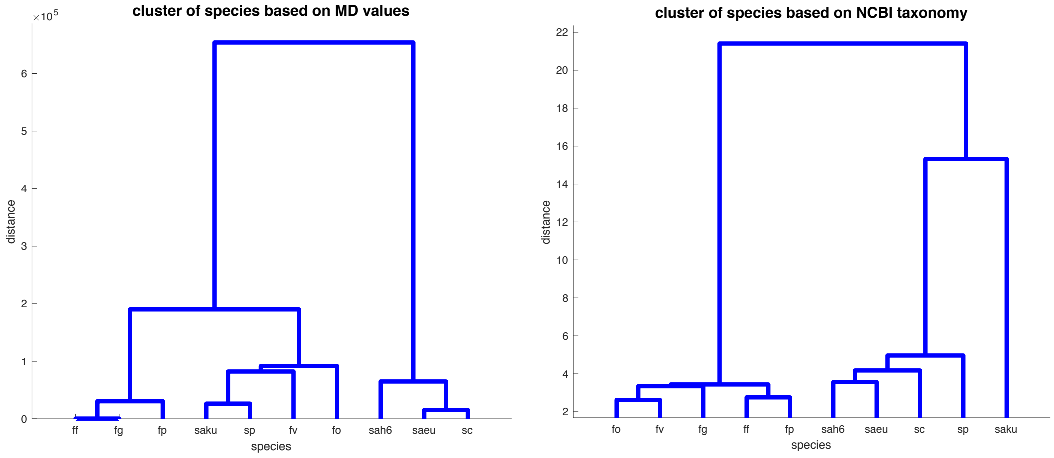


Figure 23: Two hierarchical cluster trees of 10 species based on $\mathcal{MD}$ (left) and taxonomy(right). Either cluster tree has the x-axis of the species names with abbreviations, and y-axis of the distances between nodes. Comparison between these two cluster trees we can find similar structure patterns such as *S.arboricola*, *S.eubayanus*, *S.cerevisiae* group together in both trees, and *F.fujikuroi*, *F.poae* and *F.graminearum* stay in the closest group in both trees.

First we generated the CUB cluster tree based on $\mathcal{MD}$ of these 10 species, and then generated phylogenetic tree based on the taxonomy of these 10 species. Under this setting, CUB cluster tree has 10 input variables, and each variable has 18 attributes represented by $\mathcal{MD}$; phylogenetic tree has 10 input variables and each variable has 1 attribute represented by taxonomy. $\mathcal{MD}$ represents the CUB attributes of species, and taxonomic ID represents the evolutionary distance between species. Comparison between these two cluster trees aims to investigate the correlation between CUB and phylogenetic distance among these 10 species. Figure 23 shows great similarity between these two trees, for example in both trees *S.arboricola*, *S.eubayanus*, *S.cerevisiae* group together , and *F.fujikuroi*, *F.poae* and *F.graminearum* group together. These indicate that for these 10 species their CUB tree and phylogenetic tree resemble to some extent.

#### 4.3.3.2 Similarity between CUB Cluster Tree and Phylogenetic Tree

To further explore our findings about the visual similarity between CUB cluster tree and phylogenetic tree, we compare the obtained two hierarchical clusters in Figure 23 by the comparison method detailed in section 3.3.2.

Table 16: Cluster Similarity Between CUB Cluster Tree and Phylogenetic Tree of 10 species in orders of *Saccharomycetales* and *Hypocreales*

| Cluster Number | Similarity | Dissimilarity Confidence Interval ($\alpha$=0.05) |
|---|---|---|
| 2 | 0.6390 | [0.3508, 0.6229] |
| 4 | 0.5547 | [0.0258, 0.4550] |
| 5 | 0.4558 | [-0.0299, 0.4199] |
| 6 | 0.4243 | [-0.0812, 0.3954] |

Results are shown in Table 16, where k is the cluster numbers, $B_k$ is the similarity score, $E(B_k)$ and $\texttt{var}(\texttt{B}_\texttt{k})$ are the expected value and the variance of $B_k$ for independent clusters. If a $B_k$ value is not in the range of $E(B_k) \pm 2\sqrt{\texttt{var}(\texttt{B}_\texttt{k})}$, it means the compared trees have similarity and the obtained $B_k$ has statistical significance. $B_k$=1 means the compared trees are exactly the same, while $B_k$=0 means no similarity between the cluster trees, and a higher $B_k$ value represents higher similarity between cluster trees. To be specific in our case, when we chose k=2, we got $B_k$=0.6390. Based on the $E(B_k)$ and $\texttt{var}(\texttt{B}_\texttt{k})$ for the two uncorrelated trees, we obtain the dissimilarity confidence interval [0.3508, 0.6229] at the significance level $\alpha$=0.05. This means that if dividing the 10 species into 2 groups, cluster trees are similar based on the CUB and taxonomy features of the species. When we chose k=4, we got $B_k$=0.5547, and corresponding dissimilarity confidence interval is [0.0258, 0.4550] at the significance level $\alpha$=0.05, which means the two cluster trees also have statistical significance in similarity when each tree has 4 clusters. We investigated cluster numbers up to 10, and there are statistical significant similarity between the two types of cluster trees also for k=5 and 6. Results in Table16 demonstrates that the investigated species tend to take on similar cluster performance based on $\mathcal{MD}$ and taxonomy. Such finding suggests that for certain clustering numbers there are high correlation between codon usage bias and phylogenetic distance.

To explore whether the correlation revealed by the above 10 species from orders of *saccharomycetales* and *Hypocreales* is a universal principle among Fungi kingdom, we randomly chose 10 species spread among the 462 species in Fungi kingdom, and perform the same analysis. The similarity comparison showed that for any chosen cluster numbers there are no correlation between CUB cluster tree and phylogenetic tree.

Table 17: 10 Species with Specific Taxonomy ID

| species | taxonomy ID |
|---|---|
| *Absidia glauca* | 4829 |
| *Cyberlindnera jadinii* | 4903 |
| *Pichia kudriavzevii* | 4909 |
| *Torulaspora delbrueckii* | 4950 |
| *Zygosaccharomyces rouxii* | 4956 |
| *Leucoagaricus sp* | 1714833 |
| *Phialophora attae* | 1664694 |
| *Emmonsia sp* | 1658172 |
| *Pseudogymnoascus sp 24mn13* | 1622150 |
| *Pseudogymnoascus sp 05ny08* | 1622149 |

However correlation between CUB and phylogenetic distance showed up when we chose the 10 species as Table 17. Under this setting the phylogenetic distance between the two species groups is large, while within each species group the distance among species are close. After comparing the CUB cluster tree and phylogenetic tree of these 10 species, we find high similarity when k=3, $B_k$=0.7500 has statistical significance, which is judged from the confidence interval [0.0163, 0.5508] ($\alpha$=0.05) for uncorrelated trees.

Further we expand our input nodes to 462 species in the Fungi kingdom. We built cluster tree based on CUB feature and phylogenetic distances, then applied the cluster comparison method to the two types of cluster trees. The results are illustrated in Figure 24, from which we can see that there exist $B_k$ values with statistical significance which reveals that CUB and phylogenetic distances are correlated.

In Figure 24, $B_k$ values take on a descending trend with increasing cluster k numbers. However this is an intrinsic attribute of the Fowlkes method that for small numbers of clusters, the value is high even for independent clusterings (Wagner and Wagner (2007)).

To examine the exact value of $B_k$ with statical significance, we list them in Table 18. We find that when k=15, 16, 19, 20, the two types of cluster trees have statistical significant similarity, and the highest absolute values of similarity scores is for k=15 among all the cases, however 0.3455 is not very high. The

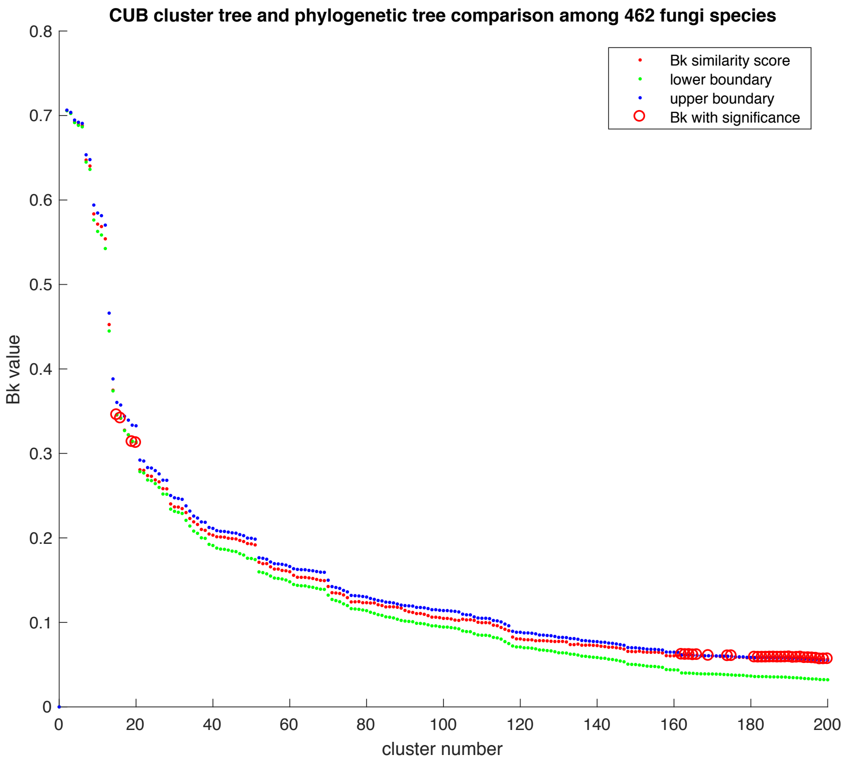


Figure 24: Similarity score of $B_k$ values between CUB cluster tree and phylogenetic tree corresponding to different cluster numbers among 462 fungal species. X-axis labels the cluster numbers of trees, y-axis labels the $B_k$ values. The red dots are similarity score between CUB cluster tree and phylogenetic tree for different cluster numbers from 1 to 200. The green and blue dots are the lower and upper boundaries of confidence intervals ($\alpha$=0.05) for Bk values of two independent cluster trees. When a red dot is outside the range of lower boundary and upper boundary, it means such red $B_k$ value has the statistical significance at $\alpha$=0.05, where the red circles mark the cases corresponding to the significant $B_k$ values representing existed similarity between trees.

Table 18: Clusters Similarity Between CUB Cluster Tree and Phylogenetic Tree of 462 species in Fungi

| Cluster Number | Similarity | Dissimilarity Confidence Interval ($\alpha$=0.05) |
|---|---|---|
| 15 | 0.3455 | [0.3462, 0.3604] |
| 16 | 0.3415 | [0.3421,0. 3573] |
| 19 | 0.3136 | [0.3146, 0.3335] |
| 20 | 0.3125 | [0.3136, 0.3327] |
| 162 | 0.0619 | [0.0402, 0.0613] |
| 163 | 0.0616 | [0.0402, 0.0621] |
| 189 | 0.0589 | [0.0353, 0.0576] |
| 190 | 0.0587 | [0.0349, 0.0573] |
| 194 | 0.0579 | [0.0336, 0.0564] |
| 195 | 0.0579 | [0.0334, 0.0562] |
| 198 | 0.0564 | [0.0324, 0.0553] |
| 200 | 0.0566 | [0.0321, 0.0552] |

reason for this may be that when large quantities of species involved in clustering, their CUB feature classification must depend on more complicated factors rather than one dominant factor of phylogenetic distance. Combining our findings among 10 species with specific phylogenetic settings, we deduce that CUB tends to correlate to phylogenetic distances for featured groups of species, where group phylogenetic distance in between is far apart while distances within groups phylogenetic distances are close.

Here we need to notice that stating $B_k$ is considered to have statistical significance is not equivalent to stating the cluster trees have high similarity. For example $B_k$=0 (where $B_k$ with great statistical significance) means that no similarity exists between the cluster trees and such dissimilarity is not happened by chance at the statistical significance level $\alpha = 0.05$ based on randomly generated uncorrelated trees.

Cluster comparison demonstrates its advantage for the correlation study between high dimensional data sets, where we have little knowledge about what correlation to be expected.

Hitherto we have introduced genome wide measure $\mathcal{MD}$ and machine learning technics for CUB correlation analysis adopting $\mathcal{MD}$, and demonstrated their application to the real fungal genomes. From the results of differences between real genomes and artificial genomes, we confirmed the validation of $\mathcal{MD}$ for it revealing stronger CUB in real species than artificial ones. From the results of SOM for input species, we found CUB patterns reproduce some features of phylogenetic relationship. From the results of SOM for input amino acids, CUB patterns to some extent reflect chemical relationship between amino acids. In addition from the results of Hierarchical Clustering we found that for remote phylogenetic groups of species their CUB features correlate to their phylogenetic distances.

# Chapter 5

# Stochastic Thermodynamics Based Model to Simulate Genome-wide CUB Pattern

In the previous three chapters we introduced a novel CUB measure. In this chapter we propose a model based on stochastic thermodynamics to investigate CUB origins.

While there is evidence for a large number of possible evolutionary drivers for CUB as stated in section 1.4, it remains unclear how the various mechanisms interact and how much they contribute to overall CUB relative to one another. Instead, a macroscopic description of the system may provide more insight. There are many precedents in science, notably in statistical physics, where simple, useful and universal laws emerge from intractable microscopic interactions. Examples, include the ideal gas law that relates macroscopic quantities to one another while ignoring individual positions and momenta of molecules, scaling laws in biology (West, Brown and Enquist (2000)), word frequencies in texts (Zörnig and Altmann (1995)), spatial structures of genomes (Cristadoro, Degli Esposti and Altmann (2018)), all of which abstract away from microscopic detail in order to arrive at robust macroscopic laws.

## 5.1 Codon Usage Bias Distribution

As shown in Figure 5, each mRNA sequence can be divided into 20 subsequences, and each subsequence is composed of synonymous codons encoding one amino acid type. The subsequence of length $L$ encoding the amino acid ($AA$) is symbolised as $S^{L,AA}$.

Assuming the number of genes in the genome is $N_g$, then the $j$-th gene $G_j$ ($j \in [1, N_g]$) has a subsequence $S_j^{AA}$ to encode the amino acid $AA$ and $S_j^{AA}$ has the length $L_j^{AA}$ ($j \in [1, N_g]$). We use the codon occurrence configuration $N_j^{AA}$ to describe the codon usage patten of the subsequence $S_j^{AA}$ in gene $G_j$. Further we adopted $N_j^{L,AA}$ distribution across the genome as the representation for amino acid specific CUB distribution across the genome.

To assist a better understanding of the meanings of the CUB distribution across the genome, we make a simple example with assumptions of short sequence lengths and a small genome size.

Consider a gene within a genome as 'GAG UUU GAA GAG UUC AUA AUU GAG AUA' . Then this gene contains the subsequence encoding amino acid GLU as 'GAG GAA GAG GAG', whose codon occurrence configuration is [3,1] and whose length is 4. Assuming searching through the genome, in total we find 4 subsequences of length 4 encoding GLU separately contained in 4 genes, and all the corresponding codon configurations are further assumed to be $N_1^{Glu} = [3, 1]$, $N_2^{Glu} = [1, 3]$, $N_3^{Glu} = [0, 4]$, $N_4^{Glu} = [3, 1]$. Then $N_1^{Glu}, N_2^{Glu}, N_3^{Glu}, N_4^{Glu}$ together reveal the CUB pattern special for amino acid chain of GLU with length 4 in the genome. When we expand the investigation of subsequence length to all the available values in the genome, all the available $N$ distributions display the codon usage pattern for amino acid GLU through out the genome.

## 5.2 Models Under Specific Selection Pressure Assumptions

To build models to explore CUB distribution, we consider codon evolution as a random walk. If seen from the perspective of codon evolution, a subsequence can be considered as a random walk system, correspondingly a codon occurrence configuration can be considered as a system state or a random walk site, thence all the available codon occurrence configurations form the whole system states or equivalently the whole random walk space. Figure 25 illustrates random walks of a subsequence with length of 4 which belongs to 2 synonymous codon family and 3 synonymous codon family separately as examples. Each random walk site corresponds to a codon occurrence configuration for the subsequence.

For simplicity, we first focus exclusively on 2 synonymous codon family, where codon evolution can be represented as a 1D random walk in discrete space and continuous time. Each site in the discrete space is a distinguished codon occurrence configuration $N$ of subsequence length $L$. For example $(n, L-n)$ defines a site in the walking space and a single synonymous codon change is sufficient to change from one site to another.

When it comes to synonymous codon families whose sizes are larger than 2, each site is connected to more than two sites in the random walk space.

When a transition from one site to another happens, only one codon is mutated for each site transition. A single codon decays with rate $r_c$, and then if there are $n_c$ codons they decay with rate $r_c n_c$, equivalently to say that the transition rate $r$ is proportional to $n_c$.

The rate of transition $r$ satisfies $r = r_c n_c$, where $r_c$ is the transition constant corresponding to the codon mutation rate, $n_c$ is the number of codons which are available to mutate. We assume that the rate of transitions where codon type 2 is converted to codon type 1 is proportional to the number of codons of codon type 2 $n_2$.

$$r(n_1, n_2, \ldots \rightarrow n_1+1, n_2-1, \ldots) \sim n_2 \tag{12}$$

where $n_i$ is the count of the $i$-th codon type.

In equilibrium, which is characterised by no net flows of probabilities, probabilities of two connected states satisfy the detailed balance condition:

$$\begin{aligned} &\pi(n_1, n_2, \ldots) r(n_1, n_2, \ldots \rightarrow n_1+1, n_2-1, \ldots) \\ =\ &\pi(n_1+1, n_2-1, \ldots) r(n_1+1, n_2-1, \ldots \rightarrow n_1, n_2, \ldots) \end{aligned} \tag{13}$$

Here $\pi$ is the occupation probability of a particular site (which means the probability to observe the system in this configuration), $r(n_1, n_2, \ldots \rightarrow n_1+1, n_2-1, \ldots)$ is the rate of a mutation from any of the $n_2$ codons of type 2 to codon type

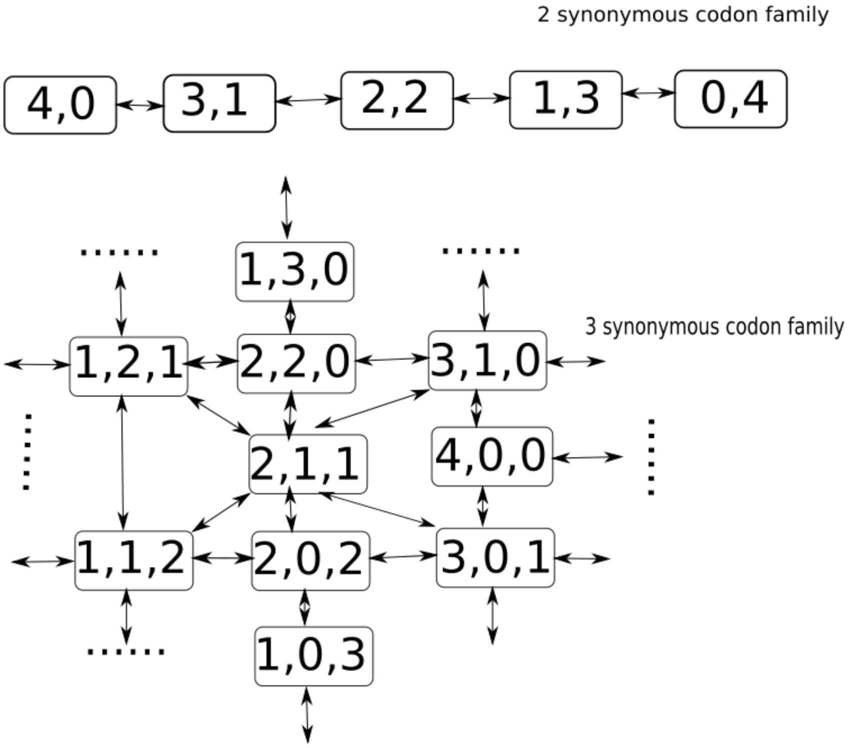


Figure 25: Illustration of the random walk for a subsequence of length 4 which belongs to 2 synonymous codon family and 3 synonymous codon family, respectively. When considering the subsequence as a system, each random walk site or system state $S = [s_1, s_2, ..., s_m]$ corresponds to a codon occurrence configuration $N = [n_1, n_2, ..., n_m]$, where m is the size of the synonymous codon family. Each entry $s_i$ in $S$ is equivalent to $n_i$ in $N$, where $i \in [1, m]$.

1, similarly $r(n_1 + 1, n_2 - 1, \ldots \rightarrow n_1, n_2, \ldots)$ is the rate of a mutation from any of the $n_1$ codons of type 1 to codon type 2. This implies that:

$$\frac{r(n_1+1, n_2-1, \ldots \rightarrow n_1, n_2, \ldots)}{r(n_1, n_2, \ldots \rightarrow n_1+1, n_2-1, \ldots)} = \frac{\pi(n_1, n_2, \ldots)}{\pi(n_1+1, n_2-1, \ldots)} = \frac{r_1(n_1+1)}{r_2 n_2} \tag{14}$$

We assume that each codon has a fixed rate of mutation to a synonymous codon. From this it follows that the rate of mutation of $n_2$ codons is proportional to $n_2$.

Next we postulate the local detail balance (Crooks (1998)):

$$\Delta E = -T \ln \left( \frac{r_+}{r_-} \right), \tag{15}$$

where $r_+$ and $r_-$ are the forwards and backwards transition rates respectively, $\Delta E$ is the abstract energy difference between the two states. This relationship then implies a dependence between the two rates, namely:

$$r_- = r_+ \exp \left( \frac{\Delta E}{T} \right) \tag{16}$$

If we assume the energy of the initial state is $E_0 = 0$, the energy of any state can be expressed by the transition rates.

In order to derive the model to simulate CUB patterns, we now have a conceptual leap: (1) Each site of the random walk has an energy $E_j$ which can be derived based on transition rates. (2) Transition rates reflect driven forces acting on codon sequences. We construct the models with different CUB driven forces by manipulating transition rates $r_+$ and $r_-$. (3) We posit that the random walker that moves between the sites is in contact with a large heat-bath that remains at a fixed temperature T. This temperature bath exchanges energy with the walker, thus enabling it to transit to sites with higher energy or lower energy by adding or extracting energy. We stress here that this idea of energy and temperature are merely conceptual devices and should not be confused with the actual physical temperature that is experienced by organisms.

We consider two models based on above concepts:

(1) *'Beanbag Model'*: codon usage is selected for at the level of the whole genome. In a random walk site, selecting which codon to mutate can be thought of as selecting beans from a bag and the probability of picking up a certain codon type is fixed independent of previous choices, thus mutation rate of any codon type is a constant.

(2) *'Sequence Level Selection Model'*: selection pressure acts at the level of the individual gene sequences. In a random walk site, mutation rate of any codon type is no longer a constant.

### 5.2.1 *Beanbag Model*

First we propose a *'Beanbag Model'* which simulates CUB pattern under the assumption that global codon usage favors certain types of synonymous codons regardless of codon positions. If each codon position has no preference for any synonymous codon type at all, we call it *Unbiased Beanbag Model*; and if there is a global codon usage preference among synonymous codon types, we call it *Biased Beanbag Model*.

*Unbiased Beanbag Model* assumes all codons have the same probability of being chosen, which means during the random walk mutation rates of all codons are the same.

To calculate the energy $E_N$ of the current state $N = [n, L-n]$ (where $L$ is the subsequence length), we start from a subsequence which has the lowest energy $E_0 = 0$ for the sake of easy calculation. For a 2 synonymous codon family, the first codon count $n$ in the codon occurrence $N$ has the same strength to define a random walk site or system state as $N$, thence $E_n$ is equivalent in description to $E_N$ where $n$ is the quantity of the first codon type in $N$. Although any configuration can be selected as an initial state, we assume that in the initial state all $L$ codons are of type 1 for the sake of easy description. A mutation can reach the next state $(n = L-1)$ by changing one of the $L$ codons of type1 to a codon of type 2. When there are no selection forces, transition rate constants are equal for every codon, then this transition happens with a forward transition rate $(r_+)$ proportional to the quantity of codon type 1 $(L)$. From this state $(n = L-1)$ the system can then move further to state $(n = L-2)$ with a forward transition rate $(r_+)$ proportional to $L-1$. Alternatively, with a backward transition rate $(r_-)$ proportional to 1 it can move back to state $(n = L)$. Altogether, the following transitions are possible:

$$(L,0) \underset{1}{\overset{L}{\rightleftharpoons}} (L-1,1) \underset{2}{\overset{L-1}{\rightleftharpoons}} \cdots \underset{n}{\overset{L-n+1}{\rightleftharpoons}} (L-n,n) \underset{n+1}{\overset{L-n}{\rightleftharpoons}} (L-n-2,n+2) \cdots \underset{L}{\overset{1}{\rightleftharpoons}} (0,L) \quad (17)$$

According to Equation 15, the energy difference $\Delta E_i$ between state $(n = i)$ and state $(n = i-1)$ is given by $\Delta E_i = E_i - E_{i-1} = -T\ln(r_+/r_-) = -T\ln\left((L-i+1)/i\right)$. Given $E_0 = 0$ the energy of state $(N)$ is :

$$\begin{aligned} E_n &= \sum_{i=1}^{n} \Delta E_i = -T \ln \left( \prod_{i=1}^{n} \frac{L-i+1}{i} \right) = \\ &= -T \ln \left( \frac{L!}{(L-n)!n!} \right) = -T \ln \binom{L}{n}. \end{aligned} \tag{18}$$

The *Biased Beanbag Model* assumes that there is an underlying bias to the mutations, namely for a random walk site which has $N$ copies of codon type 2 and $L-n$ copies of codon type 1, there exists a probability of $q$ ($q \neq 1/2$) to choose codon type 2 for mutating to type 1, and hence probability of $(1-q)$ to choose codon type 1 for mutating to type 2. Therefore the transition rate constant for codon type 1 is $q$, as well as the transition rate constant for codon type 2 is $1-q$. At such state $[L-n, n]$, the forward transition rate ($r_+$) is proportional to the probability of choosing the codon type 1 ($q$) as well as the current quantity of codon type 1 ($L-n$). Similarly $r_- = (1-q)n$. The random walk is detailed as follows:

$$(L,0) \underset{1(1-q)}{\overset{Lq}{\rightleftharpoons}} (L-1,1) \underset{2(1-q)}{\overset{(L-1)q}{\rightleftharpoons}} \cdots \underset{n(1-q)}{\overset{(L-n+1)q}{\rightleftharpoons}} (L-n,n) \cdots \underset{L(1-q)}{\overset{1\cdot q}{\rightleftharpoons}} (0,L) \tag{19}$$

Adopting Equation 15 and following the same reasoning above, we can establish the energy differences:

$$\begin{aligned} \hat{E}_0 &= 0 \\ \Delta\hat{E}_1 &= -T \ln \left( \frac{Lq}{1(1-q)} \right) \\ \Delta\hat{E}_2 &= -T \ln \left( \frac{(L-1)q}{2(1-q)} \right) \\ \Delta\hat{E}_n &= -T \ln \left( \frac{(L-n+1)q}{n(1-q)} \right) \end{aligned} \tag{20}$$

$$\begin{aligned}\hat{E}_n &= -T\ln\left(\frac{L!}{(L-n)!n!}\cdot\frac{q^n}{(1-q)^n}\right)\\ &= -T\ln\binom{L}{n}+T\ln\left(\frac{(1-q)^n}{q^n}\right)\\ &= E_n+T\ln\left(\frac{(1-q)^n}{q^n}\right) \qquad (21)\end{aligned}$$

### 5.2.2 *Sequence Level Selection (SLS) Model*

We now further propose a model assuming that there is additional selection pressures on the individual sequence beyond the global bias $q$, which is the *'Sequence Level Selection Model'* (SLS). Compared to the 'Beanbag Model' where the transition rates are proportional to corresponding codon quantities, the transition rates in SLS no longer have the linear relationship with codon quantities.

To be specific, for the 2 synonymous codon family, if the pressure on an individual sequence acts on synonymous codons with a uniform format, the rate with which codon type 2 mutates to codon type 1 is proportional to a power of the quantity of codon type 2, and vice versa. This random walk is as follows:

$$(L,0)\underset{1^\gamma}{\overset{L^\gamma}{\rightleftharpoons}}(L-1,1)\underset{2^\gamma}{\overset{(L-1)^\gamma}{\rightleftharpoons}}\cdots\underset{n^\gamma}{\overset{(L-n+1)^\gamma}{\rightleftharpoons}}(L-n,n)\cdots\underset{L^\gamma}{\overset{1^\gamma}{\rightleftharpoons}}(0,L) \qquad (22)$$

The energies then become:

$$\begin{aligned}\tilde{E}_n &= \sum_{i=1}^{n}\Delta\tilde{E}_i=-T\ln\left(\prod_{i=1}^{n}\frac{(L-i+1)^\gamma}{i^\gamma}\right)=\\ &= -T\gamma\ln\left(\frac{L!}{(L-n)!n!}\right)=\gamma E_n \qquad (23)\end{aligned}$$

which shows that if the transition rates are defined as the same exponent to the corresponding quantities, the energy format of such a model has a linear correlation with an Unbiased Beanbag Model.

Finally, we consider the most general model with different exponents assigned

Table 19: Summary of Random Walkers (2 synonymous codon family) and Corresponding Energy

| System Type | States Energy ($E_i$) | Parameters |
|---|---|---|
| Unbiased Beanbag Model | $E_n = -T \ln \binom{L}{n}$ | none |
| Biased Beanbag Model | $\hat{E}_n = E_n + T \ln \left( \frac{(1-q)^n}{q^n} \right)$ | q |
| Sequence Level Selection Full Model (SLS) | $\bar{E}_n = \xi E_n + (\gamma - \xi) \ln(n!)$ | $\gamma, \xi$ |

to different synonymous codons to arrive at the full model *'Sequence Level Selection Model'* (SLS). The random walk of SLS is as follows:

$$(L, 0) \underset{1^\gamma}{\overset{L^\xi}{\rightleftharpoons}} (L-1, 1) \underset{2^\gamma}{\overset{(L-1)^\xi}{\rightleftharpoons}} \cdots \underset{n^\gamma}{\overset{(L-n+1)^\xi}{\rightleftharpoons}} (L-n, n) \cdots \underset{L^\gamma}{\overset{1^\xi}{\rightleftharpoons}} (0, L) \tag{24}$$

This changes the energy in the following way:

$$\begin{aligned} \bar{E}_n &= \sum_{i=1}^{n} \Delta \bar{E}_i = -T \ln \left( \prod_{i=1}^{n} \frac{(L-i+1)^\xi}{i^\gamma} \right) \\ &= -T\xi \ln \binom{L}{n} + T(\gamma - \xi) \ln(n!) \\ &= \xi E_n + T(\gamma - \xi) \ln(n!) \end{aligned} \tag{25}$$

### 5.2.3 Biological Meaning of *Beanbag Model* and *Sequence Level Selection Model*

We summarise the 'Beanbag Model' and 'Sequence Level Selection Model' in Table 19. Before proceeding, we discuss special choices for the ad-hoc parameters $\xi, \gamma$ so as to clarify their biological meaning:

#### 5.2.3.1 The Special Case of SLS Model

For $\xi = \gamma = 1$ the full model Equation 25 reduces to the Unbiased Beanbag Model exactly, representing no selection pressure and no codon usage bias at all.

Any deviation of the data from $\xi = 1$ and $\gamma = 1$ conveys there exists biased codon usage due to some driving forces.

#### 5.2.3.2 Multinomial Distribution of Random Walk Sites in the Beanbag Model

We know that Boltzmann distribution connects the probability and the energy of system states by positing that a system will be in a certain state as a function of that state's energy and the temperature of the system, shown as Equation 26.

$$P(E_i) = \frac{\exp\left(-\frac{E_i}{k_B T}\right)}{\sum_i \exp\left(-\frac{E_i}{k_B T}\right)} \tag{26}$$

According to Equation 26 we obtain that the relationship between the occupation probability of a random walk site $\pi(n_1, n_2, \ldots)$ and the energy at such state $E(n_1, n_2, \ldots)$ in equilibrium as Equation 27:

$$\begin{aligned} \pi_j &= \frac{1}{Z} \exp\left(-\frac{E_j(n_1, n_2, \ldots)}{k_B T}\right) \\ Z &= \sum_{j=1}^{S_N} \exp\left(-\frac{E_j(n_1, n_2, \ldots)}{k_B T}\right) \end{aligned} \tag{27}$$

where $S_N$ is the number of possible macro states the system could possess, $\pi_j$ corresponds to the probability to observe the system in the macro state $N_j$.

From here on we set the Boltzmann constant $k_B = 1$ for convenience.

If the distribution of the codon occurrence configuration follows the multinomial distribution, the probability of observing a particular codon occurrence configuration $N$ ($[n_1, \ldots, n_m]$) of length $L$ is calculated as Equation (Equation 28).

$$P_N = \frac{L!}{n_1! \cdots n_m!} P_1^{n_1} \cdots P_m^{n_m} \tag{28}$$

where $n_i$ is the occurrence of the $i$-th codon in configuration $N$, $P_i$ is the underlying probability to choose $i$-th codon for a codon position.

In the special case for the subsequences composed of 2 synonymous codon families, the probability of observing the codon occurrence configuration [n, L-n] follows the binomial distribution as Equation 29

$$\begin{aligned} P_n = \binom{L}{n} q^n (1-q)^{L-n}, q \neq 1/2 \\ P_n = \frac{1}{2^L}\binom{L}{n}, q = 1/2 \end{aligned} \tag{29}$$

We have derived that in the Unbiased Beanbag Model $E_n = -T \ln \binom{L}{n}$. According to Boltzmann distribution (Equation 26) the probability to observe such energy $E_n$ complies with Equation 30.

$$\begin{aligned} P(E_n) = \frac{\exp\left(-\frac{E_n}{T}\right)}{\sum_i \exp\left(-\frac{E_i}{T}\right)} = \frac{\binom{L}{n}}{Z} \\ Z = \sum_i \exp\left(-\frac{E_i}{T}\right) \end{aligned} \tag{30}$$

$P(E_n)$ in Equation 30 shows the probability of observing the energy corresponding to the system state defined by $n$ derived from the random walk in the Unbiased Beanbag Model, and $P_n$ in Equation 29 shows the probability of observing the system state defined by $n$ directly calculated from the binomial distribution (q=1/2). Comparing the results from the two equations, we found the same state-related-term $\binom{L}{n}$, which reveals the binomial distribution property of the Unbiased Beanbag Model. This is consistent with the notion that without any selection pressure, the codon occurrence configuration follows a binomial distribution q=1/2; meanwhile if seen from the perspective of the random walk the codon transition rate constants are equal to each synonymous codon type.

Similarly we have derived that in the Biased Beanbag Model the system energy $\hat{E}_n = -T \ln \left[\binom{L}{n}(\frac{q}{1-q})^n\right]$ corresponds to the system state defined by $n$. According to Boltzmann distribution (Equation 26) the probability of observing such energy $\hat{E}_n$ is calculated as Equation 31

$$\begin{aligned} P(\hat{E}_n) = \frac{\exp\left(-\frac{\hat{E}_n}{T}\right)}{\sum_i \exp\left(-\frac{\hat{E}_i}{T}\right)} = \frac{\binom{L}{n}(\frac{q}{1-q})^n}{Z} \\ Z = \sum_i \exp\left(-\frac{\hat{E}_i}{T}\right) \end{aligned} \tag{31}$$

$P(\hat{E}_n)$ derived from Biased Beanbag Model has the state-related-term $\binom{L}{n}(\frac{q}{1-q})^n$ which is the same as $P_n$ in Equation 29 ($q \neq 1/2$). This reveals that the random walk sites in the Biased Beanbag Model comply with binomial distribution ($q \neq 1/2$).

In the Beanbag Model, the codon selection procedure leads to a multinomial distribution of $N^{L,AA}$ of subsequences. Whereas in the SLS model, selection of a certain codon type no longer has a consistent underlying probability among different genes, and $N^{L,AA}$ of subsequences do not follow multinomial distribution.

## 5.3 Methods of Investigation in CUB Origins Adopting ***Beanbag Model*** and SLS model

Next we describe how we fit the models to the prepared data to explore CUB origins. First we introduce how to prepare the data for fitting, then we describe fitting itself.

Datasets for fitting use the same resources as CUB measures, which include the same genome FASTA files, datasets of codon occurrence configurations, global codon usage tables (reference to section 3.2).

### 5.3.1 Datasets Generated to Fit the Models

For a subsequence of a particular length $L^{AA}$, accessible codon occurrence configurations constitute a space $\mathcal{N}$. Assuming $N_j$ is the $j$-th element in $\mathcal{N}$, when the frequency of $N_j$ is derived based on the observations in real genome, we call this 'empirical frequency'. If $N_j$ follows a multinomial distribution, the empirical frequency of $N_j$ should match the multinomial distribution probability $P_N^{AA,g}$ of observing such $N_j$.

For each genome, we produced the dataset containing multinomial distribution probability and empirical frequency of observing the corresponding codon occurrence configuration for each subsequence. Empirical frequencies are displayed in groups of amino acids, and within each amino acid group they rank in the order of increasing subsequence lengths.

Take the species *Saccharomyces cerevisiae* as an example, the multinomial

Table 20: Format of Datasets for Multinomial Distribution Probability and Empirical Frequency

| AA | $L^{\mathrm{AA,g}}$ | $P_N^{AA,g}$ | Empirical Frequency | GeneId | Codon1 |
|---|---|---|---|---|---|
| E | 1 | 5.766008e-01 | 28 | 275 | 1 |
| E | 1 | 5.766008e-01 | 28 | 298 | 1 |
| E | 1 | 4.233992e-01 | 25 | 323 | 0 |
| ... | ... | ... | ... | ... | ... |
| E | 2 | 4.882646e-01 | 37 | 6665 | 1 |
| E | 2 | 4.882646e-01 | 37 | 6731 | 1 |
| E | 3 | 7.590147e-02 | 15 | 85 | 0 |
| E | 3 | 4.223007e-01 | 33 | 130 | 2 |
| E | 3 | 7.590147e-02 | 15 | 364 | 0 |
| ... | ... | ... | ... | ... | ... |
| E | 290 | 2.098576e-02 | 1 | 2886 | 178 |
| E | 395 | 4.059443e-02 | 1 | 3972 | 228 |
| H | 1 | 6.099647e-01 | 152 | 29 | 0 |
| H | 1 | 6.099647e-01 | 152 | 54 | 0 |
| ... | ... | ... | ... | ... | ... |

distribution probabilities and empirical frequencies of observing the codon occurrence configurations in the real genome are displayed with the format in Table 20:

This table contains information about all the subsequences (with different lengths) encoding 18 different amino acids within each gene throughout the whole genome. Symbols in the header sequentially represent 'amino acid type', 'subsequence length', 'multinomial distribution probability to observe the codon occurrence configuration for such subsequence', 'empirical occurrence of such codon occurrence configuration within the genome', 'gene index within the genome', 'the first synonymous codon quantity according to the codon occurrence configuration'.

For example the line '$\mathtt{E, 3, 7.590147e - 02, 15, 85, 0}$' means:

(1) The 85th gene in the genome has 3 codons to code amino acid GLU(E) (namely the length of the subsequence to code E in this gene is 3);

(2) The first codon 'GAG' has 0 copy thus the the second codon 'GAA' has 3 copies, correspondingly the codon occurrence configuration for this subsequence is [0,3]. In the procedures to produce datasets, synonymous codon types have consistent display order in their own synonymous codon families;

(3) We searched the underlying global codon usage pattern of 'GAG' and 'GAA' in *Saccharomyces cerevisiae* from the codon usage table for Fungi kingdom, and found the synonymous codon usage ratio between 'GAG' and 'GAA' is 0.2996:0.7004, which serve as $P_1 = 0.2996$ and $P_2 = 0.7004$ in Equation 32;

(4) Based on the underlying codon usage probability [0.2996,0.7004] and the codon occurrence configuration [0,3], the multinomial distribution probability for such configuration [0,3] is 7.590147e-02 calculated according to Equation 32;

(5) We searched for subsequences coding GLU(E) which has the same codon occurrence configuration [0,3] throughout the genome of *Saccharomyces cerevisiae*, and found 85 cases of [0,3].

Next we introduce how we obtain multinomial distribution probability for each observed codon occurrence configuration.

Different from our CUB measurement which assumes that the underlying synonymous codon usage probabilities are uniform, datasets adopted to fit our model were generated under the assumption that the underlying synonymous codon usage probabilities are consistent with the global codon usage table.

When accepting global codon usage table $P^{AA} = [P_1, P_2, ...P_i, ..., P_m]$ as the underlying probabilities to select synonymous codons to possess a codon position, each codon occurrence configuration $N = [n_1, n_2, ...n_i^{\mathrm{AA},g}, ...n_m]$ has a corresponding multinomial distribution probability $P_N^{AA,g}$, calculated as Equation 32.

$$\begin{aligned} P_N^{AA,g} = \frac{L^{AA,g}!}{n_1!n_2!...n_m!} P_1^{n_1} P_2^{n_2} ... P_m^{n_m} \\ \sum_{i=1}^{m} n_i = L^{AA,g} \\ \sum_{i=1}^{m} P_i = 1 \\ m = |C^{AA}| \end{aligned} \tag{32}$$

If the underlying $P_i \neq P_j$ for any $i \neq j$, $P_N^{AA,g}$ and $N = [n_1, \ldots, n_{|C^{AA}|}]$ have one-to-one correspondence, the distribution of $N$ and the distribution of $P_N^{AA,g}$ are functionally equivalent to describe the CUB distribution.

Table 21: Dataset Contain Multinomial Distribution Probability and Empirical Frequency

| Dataset | Genome Type | $P^{AA}$ |
|---|---|---|
| Tb | Real Genome | Biased Synonymous Codon Usage |
| Tab | Substituted Control Genome | Biased Synonymous Codon Usage |

### 5.3.2 Two Types of Datasets

Datasets containing multinomial distribution probability and empirical frequency are generated based on real and control genomes under assumptions of different underlying global codon usage preferences $P^{AA}$. Displayed as Table 21, there are two kinds of datasets for fitting which are Tb, and Tab ('T' represents 'table', 'a' represents 'artificial', 'b' represents 'bias' ):

(1) Tb: $P_N^{AA,g}$ are calculated based on the real genome, and the biased underlying $P^{AA}$ is consistent with global codon usage table.

(1) Tab: $P_N^{AA,g}$ are calculated based on the artificial genome, and the biased underlying $P^{AA}$ is consistent with global codon usage table. The artificial genome is obtained by replacing codon with its synonymous codons with biased probability according to codon usage table (detailed procedure refers to section 3.2.5).

$P_N^{AA,g}$ and $P^{AA}$ have the meanings explained by Equation 32. By the approaches of construction, the artificial genomes have the property that codon occurrence configuration of the subsequences complies with multinomial distribution.

All the 'Tb' and 'Tab' have the same structure of data, which means regression approaches are the same for these two types of genomes.

### 5.3.3 Empirical Energy

For each set of subsequences corresponding to a particular length and amino acid, we next define an empirical energy $\mathcal{E}$ as Equation 33

$$\mathcal{E}_{n_1,\ldots,n_{|C^{AA}|}} := T \ln\left(\frac{n_{n_1,\ldots,n_{|C^{AA}|}}}{S_N}\right), \tag{33}$$

where $n_{n_1,\ldots,k_{|C^{AA}|}}$ is the occurrence of configuration N ($[n_1, \ldots, n_{|C^{AA}|}]$) in the genome, $S_N$ is the number of subsequences encoding amino acid $AA$ with length $L$ in the genome where $L = \sum_{i=1}^{|C^{AA}|} n_i$ .

Our global codon usage table contains synonymous codon usage proportions for each investigated amino acid in all the investigated species. After automatic searching through all the elements of the global codon usage table, we did not find any case of $P_i = P_j$ $(i \neq j)$, therefore $P_N^{AA,g}$ distribution is a monotone function (Alam (1970)) and is equivalent to present the distribution of macro states $N$ in the investigated genomes. This means that we can obtain the empirical energy according to Equation 34.

$$\mathcal{E}_{n_1,\ldots,n_{|C^{AA}|}} := T \ln \left( \frac{n_{P_N^{AA,g}}}{S_N} \right) \tag{34}$$

where $n_{P_N^{AA,g}}$ is the count of a certain multinomial distribution probability value of $P_N^{AA,g}$ within the genome. $P_N^{AA,g}$ is calculated according to Equation 32 based on codon occurrence configuration N ($[n_1, \ldots, n_{|C^{AA}|}]$) for amino acid $AA$.

The empirical energy calculation depends on obtaining the occurrence of each particular macro state across the whole genome, however counting the macro states becomes computational onerous as the sequence length and synonymous codon choices increase. To overcome such difficulty, we found that when the underlying probability vector has no equal entries, the multinomial distribution probability is a monotone function (Alam (1970)) mapping from macro states to their corresponding probabilities, therefore counting the same multinomial distribution probability is equivalently counting the same macro state.

If there is no sequence level selection, the distribution of codon occurrence configurations within the observed genome should comply with a multinomial distribution, and hence $\mathcal{E}^L_{n_1,n_2,\ldots,n_{|C^{AA}|}}$ and $\hat{E}^L_{n_1,n_2,\ldots,n_{|C^{AA}|}}$ should have a linear relationship ($\hat{E}$ corresponds to the Biased Beanbag Model with the attribute that codon occurrence configuration follows multinomial distribution).

### 5.3.4 Variable Pairs to be Fitted

Equation 32 and 34 demonstrate that each subsequence corresponds to a pair of variables $[n_1, \mathcal{E}_{k_1,\ldots,k_{|C^{AA}|}}]$, where $n_1$ is the copy number of the first codon type in its configuration $N = [n_1, n_2, ..., n_{|C^{AA}|}]$. Here we focus on the family of amino acid $AA$ that are encoded by 2 synonymous codon. Thus all the subsequences which encode the same amino acid and have the same length in all the genes of a genome form the paired variable space.

Models will be fitted to the paired domains of the two variables $n_1$ and $\mathcal{E}_{n_1,\ldots,n_{|C^{AA}|}}$. The independent variable $n_1$ is the first synonymous codon occurrence of a codon occurrence configuration $N$, and the response variable $\mathcal{E}$ is the empirical energy corresponding to such N $([n_1, \ldots, n_{|C^{AA}|}])$.

### 5.3.5 Nonlinear Regression Functions to Fit the Paired Variable Domains

We have proposed two models to simulate codon usage distribution by way of depicting the relationship between the state energy and the state of codon occurrence configuration. The Biased Beanbag Model assumes there is only global codon usage selection happening at the codon type level, which is represented by parameter 'q' as the preference for the first codon type. The full SLS Model assumes that besides a global selection preference for a certain codon type, there is also sequence level selection pressure influencing codon usage pattern, which are represented by parameters '$\xi$', and '$\gamma$', where '$\xi = 1, \gamma = 1$' describes a special case that there is no selection pressure at all.

When fitting the Biased Beanbag Model to variable domains of $[n_1, \mathcal{E}_{k_1,\ldots,k_{|C^{AA}|}}]$ derived from Tb' and 'Tab' datasets, the nonlinear regression function is $\hat{E}_n = -T \ln \binom{L}{n} + T \ln \left( \frac{(1-q)^n}{q^n} \right)$. To perform the regression, $n_1$ is the independent variable, $\mathcal{E}(n_1)$ is the response variable, $\hat{E}_(n_1)$ is the predicted variable, and 'q' is the parameter required for estimation.

When fitting the SLS Full Model to variable domains $[n_1, \mathcal{E}_{k_1,\ldots,k_{|C^{AA}|}}]$ derived from 'Tb' and 'Tab' datasets, the nonlinear regression function is $\bar{E}_n = -T\xi \ln \binom{L}{n} + T(\gamma - \xi)\ln(n!)$. To perform the regression, $n_1$ is the independent variable, $\mathcal{E}(n_1)$ is the response variable, $\bar{E}_(n_1)$ is the predicted variable, and '$\xi, \gamma$' are the parameters required for estimation.

### 5.3.6 Nonlinear Regression Procedure

Mean residual (MR) is taken as the goodness assessment for a nonlinear fitting, which is calculated as MR$= \sum_{i=1}^{S_N} \left( \dot{E}(n_1^i) - \mathcal{E}(n_1^i) \right)$, where $n_1$ is the independent variable (first codon count in configuration $N$), $\dot{E}(n_1)$ is the predicted variable (predicted energy), $\mathcal{E}(n_1)$ is the response variable (empirical energy), $S_N$ is the number of fitted variable pairs. We perform nonlinear regression by Maple's nonlinear fit as follows:

(1)We fitted the Biased Beanbag Model function to the empirical data [$n_1$, $\mathcal{E}_{n_1,\ldots,n_{|CAA|}}$] and estimate the parameters $q$; also fitted the SLS full model to the empirical data [$n_1$, $\mathcal{E}_{n_1,\ldots,n_{|CAA|}}$] and estimate parameters [$\xi$, $\gamma$].

(2)For each species, we did this for all available genes and each amino acid separately for groups of subsequences of length 5-15.

(3) The nonlinear fit requires an initial guess for the parameters to be estimated. For Biased Beanbag Model, we use $q_0$ matching global codon usage table as the starting point; for SLS full model, we use $(\xi_0, \gamma_0)$ as the starting point, where $\xi_0 = 1$ and $\gamma_0 = 1$.

(4)After fitting we evaluated the mean residual as a quality measure of the fit. If this value was above the threshold of 0.000999 then we repeated the fit with a random initial guess for the initial parameters up to 100 times until the residual value is lower than the threshold.

(5)We then record the fitting parameters and MR thus obtained.

## 5.4 Results of Regression Adopting *Beanbag Model* and *SLS Model*

We selected subsequences with lengths $5 \leq L^{AA,g} \leq 15$, for the reason that there are very few different codon occurrence configurations to consider for subsequences of length $< 5$, and statistical errors become overly large due to distinguished available subsequence sample reduces quickly with increasing subsequence length $> 15$.

Considering the complexity of random walk model for more than 2 synonymous codon families, we first focus on nine amino acids encoded by 2 synonymous codons.

### 5.4.1 *Biased Beanbag Model* can be fitted to Tb Datasets

Variable paris [$n_1$, $\mathcal{E}_{n_1,\ldots,k_{|CAA|}}$] in Tb dataset can be fitted to the Biased Beanbag Model shown in Figure 26(a). Doing this for all subsequences results in a distribution of mean residual between $\exp(-4)$ and $\exp(-9)$ peaking at about $\exp(-7)$.

The only fitting parameter in the model is the global codon usage bias $q$.

### 5.4.2 *SLS Model* Fits Tb Datasets Better

As a comparison we also fitted the full model to the variable pairs $[n_1, \mathcal{E}_{n_1,\ldots,k_{|CAA|}}]$ in the Tb datasets, thus obtaining mean residual and estimated values for the parameters $\xi$ and $\gamma$.

For all our datasets, the typical values of the parameters $\gamma$ and $\xi$ are small and positive with 96.39% of the fits resulting in $0 < \gamma, \xi < 2$. The quality of the fits can be quantified by comparing the MR obtained from fitting the SLS full model with those obtained from fitting the Biased Beanbag Model. See Figure 26 for a comparison of the MR distributions. This indicates that the former is a better description of the data in the sense that the distribution of MR is shifted to the left towards smaller values. The median for the residuals of the full model is 0.0002850, about 3 times smaller than the corresponding value for the Biased Beanbag Model fits, which is 0.000845.

### 5.4.3 Meaning of the Better Fit of *SLS Model* to Tb Datasets

The better fit of the SLS full model to Tb dataset could be merely a reflection of the fact that it has more parameters than the Biased Beanbag Model. We therefore prepared a control set of MR distributions.

The random substituted genome performs as the control group representing the genome only with the global preference for certain codon types, without sequence level selection pressure. Because of the way this control genome was constructed, it should comply with the Biased Beanbag Model exactly.

Fitting both the SLS full model and the Biased Beanbag Model to the control datasets Tab, results in MR that are visually indistinguishable from one another as shown in Figure 26a. This conveys that the SLS full model can be equivalent to Biased Beanbag Model.

The quality of the fit of the Biased Beanbag Model to the the control-set can be viewed as a benchmark for the best MR that can be obtained given the statistical error inherent in the dataset. An inspection of the histogram in Figure 26a reveals that the distribution of MR obtained from fitting the full model to the real data only minimally shifts to the right of this optimal benchmark, which demonstrates a good fit. MR obtained from fitting the Biased Beanbag Model to the real data obviously shifts to the right of the optimal benchmark. This leads to the conclusion that the SLS full model captures almost all of the underlying variations of the real data. The Biased Beanbag Model is not sufficient to explain how codons are distributed across the genomes in fungi kingdom. Instead, it is necessary to postulate sequence-level selection in order to account for the distribution of CUB across genomes.

### 5.4.4 Defining Distance as a Measure of Selection Pressure

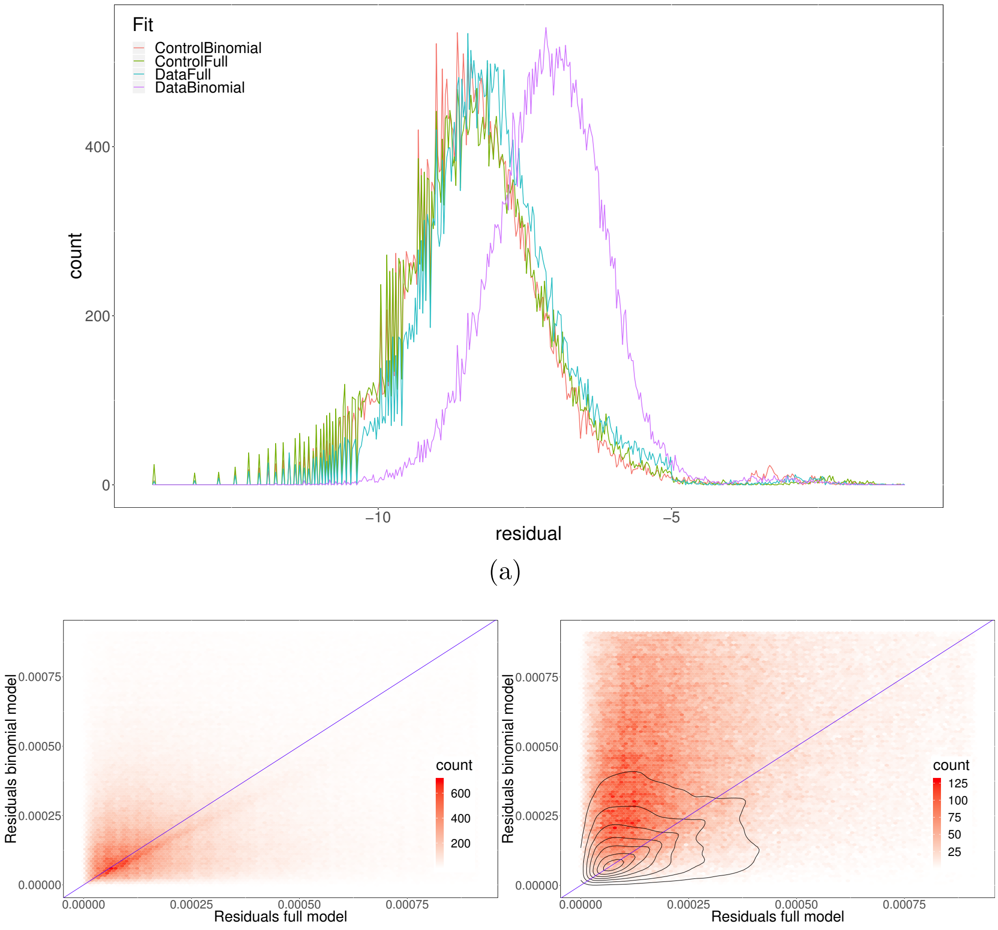
A different perspective of looking at parameters $\xi, \gamma$ space (see Figure 27b) reveals that the fits of control data concentrate in a smaller part of parameter space than the fits to the real data.

Based on the full model we now propose Euclidean distance between the observed case and the no-selection case in $\xi, \gamma$ space. This no-selection case corresponds to $\xi = \gamma = 1$ exactly and any deviation from that indicates a selection pressure.

$$\mathcal{D} := \sqrt{(1-\xi)^2 + (1-\gamma)^2} \qquad \text{(Selection pressure)}$$

To apply $\mathcal{D}$ to characterise selection pressure at the sequences level, both in Tb datasets and Tab datasets for fungi kingdom, distributions of $\mathcal{D}$ are shown in Figure 29. Real genomes universally have larger $\mathcal{D}$ values compared to control groups, which again proves that global selection pressure is not sufficient enough to explain CUB patterns in real genomes.

To apply $\mathcal{D}$ to spot selection pressure at the sequence level, we selected all subsequences where the global codon usage bias towards codon 1 is between 0.495 and 0.5 in the fungi kingdom. The distribution of $\mathcal{D}$ is shown in Figure 30. The Biased Beanbag Model would predict that these subsequences have a distance of 0. It is apparent that there are many examples of subsequences that have no global bias, but at the same time subject to a SLS pressure, as evidenced by a distance that is different from 0.

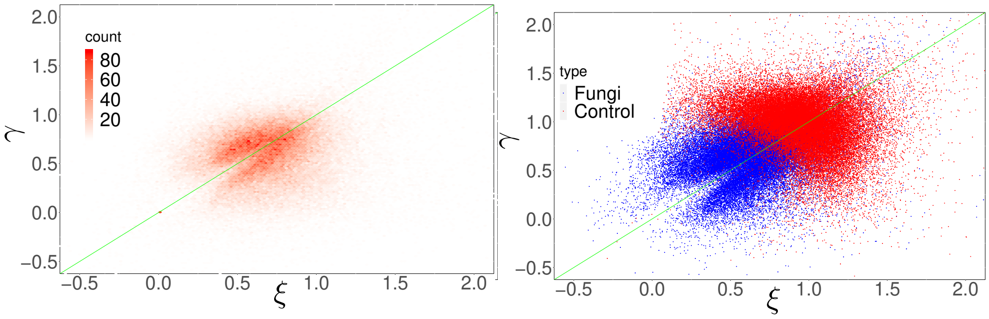


(a)

(b)

(c)

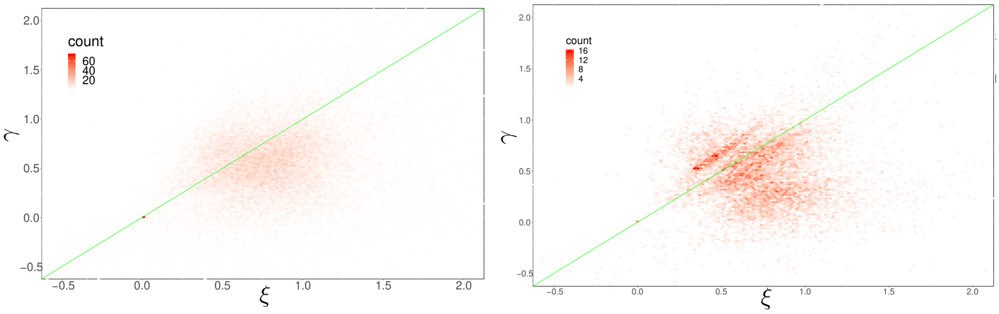
Figure 26: (a) Histogram for the MR obtained from fitting the Biased Beanbag Model and the SLS full model to both the Tb real data and Tab control data. The $x$-axis is shown on a logarithmic scale. The distribution of the MR of the Biased Beanbag Model fitted to real data is clearly shifted to the right compared to the fit of the full model, suggesting that the latter is a better fit on the whole. On the other hand, the fitting results of control data display that the MR distribution of the full model overlap with the Biased Beanbag Model. (b) Comparing the MR from the full model to those of the Biased Beanbag Model, the plot shows the density of points for the Tab datasets. The area above the diagonal indicates subsequences where the full model is a better fit than the Biased Beanbag Model. Points on the diagonal indicate that both models fit the subsequence equally well. (c) Same comparison, but for Tb real datasets. The contour lines indicate the density of the control data in (b) for comparison.



(a)

(b)

Figure 27: (a) The density of fitted parameters $\xi$ and $\gamma$ for each of 2 synonymous codon family for all the 462 fungal species in our dataset. We are limiting ourselves to fits with MR < 0.0009999. The estimated parameters largely concentrate in the interval of $[0, 1.5]$. (b) Comparison between the estimated parameters obtained from the Tb real genome datasets (red) and estimated parameters obtained from Tab the control genome datasets (blue). The plot shows actual points rather than density.



(a)

(b)

Figure 28: The fitted values of parameters $\xi$ and $\gamma$ for each of the 2-codon amino acids for bacteria and protists. The graphs show heatplots that summarise the density of points in the area. Red indicates a high density of points. We are limiting ourselves to those amino acid subsequences that have a sub-length of 15.

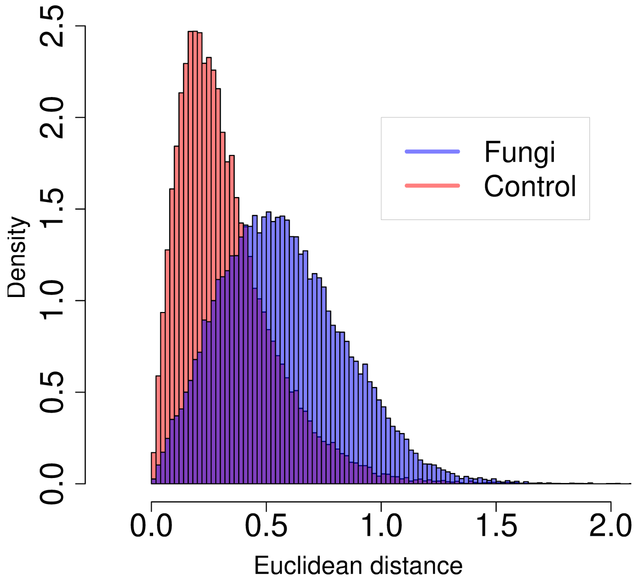


Figure 29: The distribution of distances $\mathcal{D}$. $\mathcal{D}$ calculated from the control data (red) clearly has a smaller distance on the whole than from the real data (blue), indicating that considering only the global codon usage bias underestimates the selection pressure in real genomes.

#### 5.4.4.1 $\mathcal{D}$ reveals amino acid-specific patterns of codon selection pressure

An advantage of using $\mathcal{D}$ over other measures of codon usage bias is that it lends itself to detecting differences in selection pressure in different subsequence sets. By way of example, we compared how $\mathcal{D}$ differs for different amino acids in the fungal kingdom. Initial visual inspection of the dataset revealed that, as a general pattern, most amino acids in the same organism behave similarly in terms of $\mathcal{D}$, suggesting that they experience similar selective forces. There are, however, also exceptions to this pattern. Fig. 31 reveals that atypically stronger or weaker selection for particular amino acids is an evolutionary feature that is linked to taxonomic groups. In this analysis, we defined atypical selection as a $\mathcal{D}$ value that is more than 2 standard deviations above or below the average $\mathcal{D}$ value for that organism.

Particularly notable patterns include the *Sordariomycetes* group where the amino acid phenylalanine (F) shows atypically strong codon selection (higher than

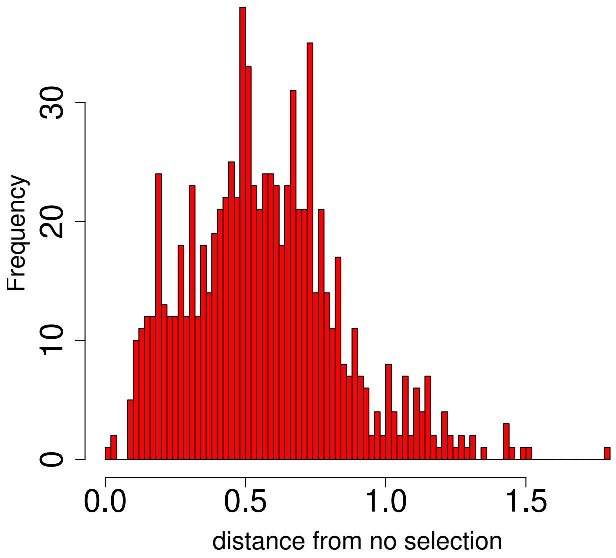


Figure 30: Distribution of distances $\mathcal{D}$ in genomes that have no global CUB. $\mathcal{D}$ value different from 0 is the evidence that sequence level selection exists in the genomes although with no global codon usage bias.

average $\mathcal{D}$ values) in most species. Interestingly, the pattern is reversed in the *Agaricomycotina* group where selective pressure on phenylalanine codon usage is weaker than for other codons, and in the *Leotiomycetes* group the selection force on phenylalanine codon usage is similar to that of other codons. Some of the observed patterns are highly interesting. For example glutamic acid (E) and aspartic acid (D) are physically very similar, negatively charged amino acids that can frequently be substituted in evolution. Nevertheless in these analyses they show quite distinct behaviour in terms of codon usage selection. The fact that these patterns have remained hidden throughout decades of analysis illustrates the usefulness of $\mathcal{D}$ as a measure of selection acting on codon usage bias.

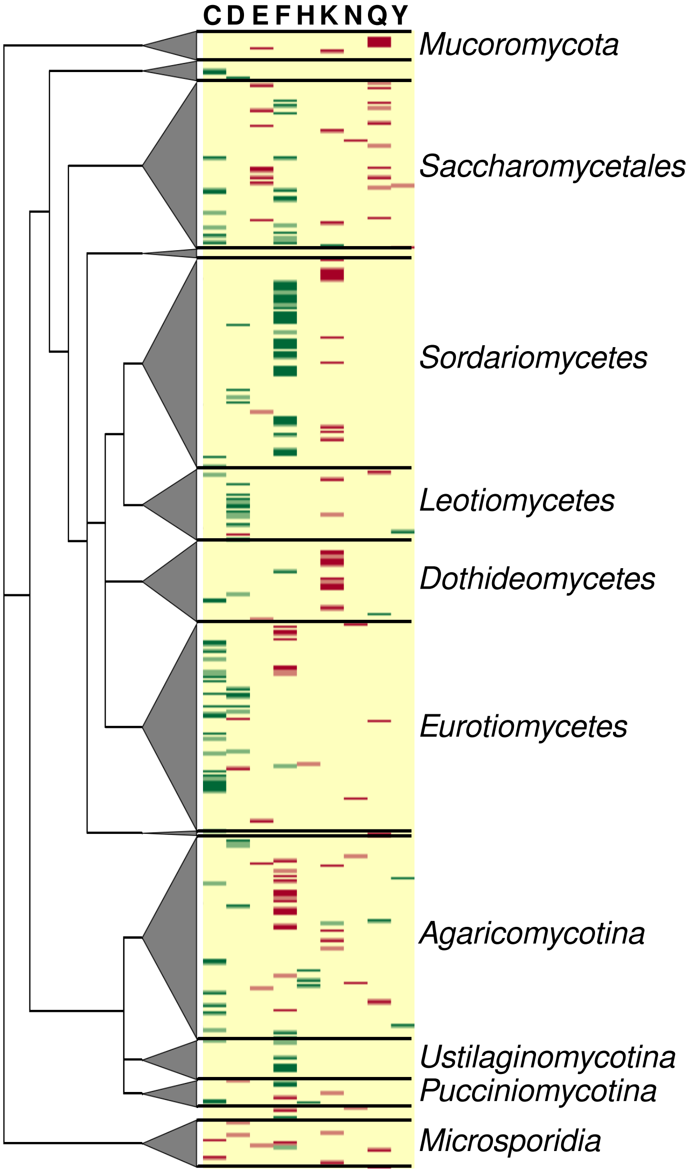


Figure 31: Amino acid-specific patterns of codon usage bias in fungal genomes. Average $\mathcal{D}$ values were calculated for all subsequences for each amino acid and each genome. Amino acids are highlighted if their $\mathcal{D}$ value was more than $2\sigma$ above (green) or below (red) the median $\mathcal{D}$ for that species. In other words, red and green highlights indicate amino acids that are under atypical selection compared to other amino acids in the same species. Species were ordered according to the taxonomic hierarchy in NCBI taxonomy, and taxonomic groups represented with larger numbers of genomes are indicated.

# Chapter 6

# Conclusion

We proposed novel methods for CUB analysis and also applied them to in total 1047 organisms among 3 kingdoms of Fungi, Bacteria and Protist, and our results reveal hidden CUB patterns across species.

## 6.1 Contributions of the Novel CUB Measure

Our CUB measure does not require any external biological reference dataset but only the nature of the codon sequences themselves, and hence it is applicable to any genome regardless of the degree of the required knowledge available. Furthermore it conceptualises codon evolution in a mathematical algorithm and has high computational efficiency.

Our CUB measures $Sn$, $MD$ and $\mathcal{MD}$ quantify CUB at different levels of a sequence, an amino acid and a genome.

$Sn$ measures CUB for a particular amino acid in a specific sequence. It demonstrates the deviation of a codon sequence from the observed state to the most probable state. It is reasonable to assume that without any driving force codon usage configuration should distribute around the maximum likely state, therefore $Sn$ has the meaning to represent the force which drives codon sequences deviating from the total random usage. CUB patterns of $Sn$ derived from orthologs for functional grouped genes revealed relationship between CUB and gene functions. Protein abundance and CUB correlation study adopting $Sn$ demonstrated that with the increasing demand for protein abundance in cell, genes decrease their lengths but keep their CUB at a high level, and highly expressed genes tend to be short but highly biased.

$MD$ measures CUB for a particular amino acid type at the whole genome level. By way of comparing $Sn$ with theoretical expected value of $Sn$ ($\overline{Sn}$) for each individual length, MD combined $Sn$ values into a genome wide amino acid specific CUB assessment parameter, which maintains different CUB information carried by individual gene and also tackles well with the intrinsic variations of empirical $\overline{Sn}$ resulting from different sequence lengths.

$\mathcal{MD}$ is a genome wide CUB measure which combines CUB information for all the genes and all the amino acids throughout the genome. It takes the form of a vector containing $MD$ values of all the amino acids. Self Organising Map and Hierarchical Cluster treat the high dimensional CUB assessment parameter $\mathcal{MD}$ as a whole feature for a species, by which way it minimises the CUB information loss or distortion if by simple linear combination of amino acid specific or sequence specific CUB measures. Results of Self Organising Map analysis revealed that CUB patterns at amino acid level relate to chemical properties of amino acids and also CUB patterns among species reproduce some features of phylogenies. Similarity quantification derived from comparison between CUB cluster tree and phylogenetic tree of grouped species revealed that CUB has correlation with phylogenetic distances when phylogenetic distance between groups is large but within groups species are close.

The proposed CUB measure does not require external reference sets, has validated meaning of strength of driving forces at the sequence level, and has reasonable combinations of sequence specific and amino acid specific measures into a parameter in a genome wise manner. It diminishes the sequence length impact on CUB measure, maintains different CUB information carried by individual gene and individual amino acid type, and makes the correlation analysis between high dimensional features possible. None of the published CUB measures possess all these advantages at the same time.

## 6.2 Contributions of Sequence Selection Model of CUB

Among the published results about CUB origins, many different drivers of codon selection have been described ranging from intrinsic genomic features to elements involved in the key stages of protein synthesis, but perhaps many remain to be discovered. However at present there is no consensus on the evolutionary drivers of codon usage bias.

Listing all the driving forces of CUB and disentangling how they act and interact is probably an intractable task, but collectively these forces seem to behave in a simple way, leading to a macroscopic description of codon usage bias. Our sequence selection model offered a new perspective on this issue based on a parsimonious expression for codon distributions across the whole genome based on concepts from statistical physics. Our work refrained from committing to a particular selection mechanism, but focused on the aggregate effect of all selection forces which are summarised by a parsimonious mathematical model with only 2 parameters. The two parameters can be directly interpreted in terms of selection forces namely as the exponents modifying the rate of synonymous mutations from one codon to another one.

Based on the 2 parameters, we derived a distance from the no-selection case, which takes into account not only the global codon usage preference but also the CUB distribution across the genome. In the special case of no global codon usage bias, however, we showed that in fact even those subsequences do show signatures of selection .

We apply the model to genomic data across 3 kingdoms and find that it captures almost all aspects of the empirical data, and that it allows new insights about the evolutionary pressures that shaped codon usage. An immediate conclusion we draw from our results is that there must be significant selection pressures on codon usage bias at the level of individual gene sequences. This insight will provide an immediate impetus to new research on what these sequence-level pressures could be. There are no statistical differences of CUB distribution among fungi, bacteria and protist kingdoms, but our model makes it applicable to investigate and compare CUB distribution in higher life organisms such as plants and animals.

We limited our analysis currently to the 2 synonymous codon families. In principle, there is no theoretical difficulty to extend the model to other synonymous codon families. The binomial distribution needs to be replaced with a multinomial distribution and the full model needs to be adapted to include an extra parameter for each additional synonymous codon type. In practice, the analysis becomes problematic because based on available codon occurrence configurations from the genome data, observed sample size is not large enough to retrieve a reasonable empirical energy. With more codons the number of possible subsequence compositions grow quickly, but the number of available observed subsequences does not. As a consequence, there are fewer examples per configuration which increases the statistical error.

## 6.3 Potential Applications

We have performed correlation analysis between CUB and protein abundances in cells based on $Sn$. This demonstrates an example for correlation analysis based on $Sn$ between sequence specific CUB and other factors of interest. Such factors can be tRNA abundances, ribosome binding rates, tRNA aminoacylation rates etc..

In addition we used Self Organising Map and Hierarchical Clustering based on $MD$ and $\mathcal{MD}$ to perform correlation analysis between high dimensional CUB features and phylogenetic distances. This illustrated a feasible method to make correlation study between any high dimensional CUB features and any interested factors, and the interested factors can also be high dimensional vectors.

By constructing feature space based on our multilevel CUB measures, and the other feature space based on any combination of interested factors, the cluster similarity quantification is competent to explore correlations between CUB and any high dimensional features. This means, if the constructed feature space has specific study purpose, the results of tree comparison could convey specific correlation information.

Potential applications of our proposed measure could be studies on fungal pathogens infecting human at the species level, and gene related diseases resulting from synonymous codon usage mutation at the sequence level. We can adopt Hierarchical Clustering to probe into possible causes of diseases related to codon usage bias. Accumulation of abundant and pertinent information is necessary for the construction of features spaces, for example which species or genes to be investigated, what common or unique properties they have, such as temperature, humidity, nutrients of the species habitats; duplication rate, expression efficiency of genes causing disease and etc.. Although loads of information are required to be prepared, making contributions to health of human beings must be a meaningful and promising application of our proposed CUB measure.

# Bibliography

Adl, S. M. et al. (2012). The revised classification of eukaryotes. *Journal of eukaryotic microbiology*, 59(5), pp. 429–514.

Agashe, D. et al. (2012). Good codons, bad transcript: large reductions in gene expression and fitness arising from synonymous mutations in a key enzyme. *Molecular biology and evolution*, 30(3), pp. 549–560.

Aggarwal, C. C., Hinneburg, A. and Keim, D. A. (2001). On the surprising behavior of distance metrics in high dimensional space. In *International conference on database theory*, Springer, pp. 420–434.

Akinduko, A. A. and Mirkes, E. M. (2012). Initialization of self-organizing maps: principal components versus random initialization. a case study. *arXiv preprint arXiv:12105873*.

Alam, K. (1970). Monotonicity properties of the multinomial distribution. *The Annals of Mathematical Statistics*, pp. 315–317.

Angov, E. (2011). Codon usage: nature's roadmap to expression and folding of proteins. *Biotechnology journal*, 6(6), pp. 650–659.

Asakawa, S. et al. (1991). Strand-specific nucleotide composition bias in echinoderm and vertebrate mitochondrial genomes. *Journal of molecular evolution*, 32(6), pp. 511–520.

Badet, T. et al. (2017). Codon optimization underpins generalist parasitism in fungi. *Elife*, 6, p. e22472.

Baeza, M. et al. (2015). Codon usage and codon context bias in xanthophyllomyces dendrorhous. *BMC genomics*, 16(1), p. 293.

Behura, S. K. and Severson, D. W. (2011). Coadaptation of isoacceptor trna genes and codon usage bias for translation efficiency in aedes aegypti and anopheles gambiae. *Insect molecular biology*, 20(2), pp. 177–187.

Behura, S. K. and Severson, D. W. (2012). Comparative analysis of codon usage bias and codon context patterns between dipteran and hymenopteran sequenced genomes. *PloS one*, 7(8), p. e43111.

Behura, S. K. and Severson, D. W. (2013). Codon usage bias: causative factors, quantification methods and genome-wide patterns: with emphasis on insect genomes. *Biological Reviews*, 88(1), pp. 49–61.

Bennetzen, J. L. and Hall, B. D. (1982). Codon selection in yeast. *Journal of Biological Chemistry*, 257(6), pp. 3026–3031.

Berg, J., Tymoczko, J. and Stryer, L. (2002). Eukaryotic protein synthesis differs from prokaryotic protein synthesis primarily in translation initiation. *Biochemistry*, 5.

Birdsell, J. A. (2002). Integrating genomics, bioinformatics, and classical genetics to study the effects of recombination on genome evolution. *Molecular biology and evolution*, 19(7), pp. 1181–1197.

Bulmer, M. (1991). The selection-mutation-drift theory of synonymous codon usage. *Genetics*, 129(3), pp. 897–907.

Cannarozzi, G. M. and Schneider, A. (2012). *Codon evolution: mechanisms and models*. Oxford University Press.

Chan, P. P. and Lowe, T. M. (2008). Gtrnadb: a database of transfer rna genes detected in genomic sequence. *Nucleic acids research*, 37(suppl_1), pp. D93–D97.

Chu, D., Barnes, D. J. and von der Haar, T. (2011). The role of trna and ribosome competition in coupling the expression of different mrnas in saccharomyces cerevisiae. *Nucleic Acids Res*, 39(15), pp. 6705–14.

Clarke, B. (1970). Darwinian evolution of proteins. *Science*, 168(3934), pp. 1009–1011.

Coleman, J. R. et al. (2008). Virus attenuation by genome-scale changes in codon pair bias. *Science*, 320(5884), pp. 1784–1787.

Comeron, J. M. and Aguadé, M. (1998). An evaluation of measures of synonymous codon usage bias. *Journal of molecular evolution*, 47(3), pp. 268–274.

Cope, A. L., Hettich, R. L. and Gilchrist, M. A. (2018). Quantifying selection on codon usage in signal peptides: Gene expression and amino acid usage explain apparent selection for inefficient codons. *bioRxiv*, p. 347849.

Crick, F. H. (1968). The origin of the genetic code. *Journal of molecular biology*, 38(3), pp. 367–379.

Cristadoro, G., Degli Esposti, M. and Altmann, E. G. (2018). The common origin of symmetry and structure in genetic sequences. *Scientific reports*, 8(1), p. 15817.

Crooks, G. E. (1998). Nonequilibrium measurements of free energy differences for microscopically reversible markovian systems. *Journal of Statistical Physics*, 90(5-6), pp. 1481–1487.

Crow, J. F., Kimura, M. et al. (1970). An introduction to population genetics theory. *An introduction to population genetics theory.*

Cruz-Vera, L. R. et al. (2004). Ribosome stalling and peptidyl-trna drop-off during translational delay at aga codons. *Nucleic acids research*, 32(15), pp. 4462–4468.

Danneels, B., Pinto-Carbó, M. and Carlier, A. (2018). Patterns of nucleotide deletion and insertion inferred from bacterial pseudogenes. *Genome biology and evolution*, 10(7), pp. 1792–1802.

Dilucca, M., Cimini, G. and Giansanti, A. (2018). Essentiality, conservation, evolutionary pressure and codon bias in bacterial genomes. *Gene*, 663, pp. 178–188.

Du, M.-Z. et al. (2018). The gc content as a main factor shaping the amino acid usage during bacterial evolution process. *Frontiers in microbiology*, 9.

Duret, L. (2000). trna gene number and codon usage in the c. elegans genome are co-adapted for optimal translation of highly expressed genes. *Trends in Genetics*, 16(7), pp. 287–289.

Duret, L. (2002). Evolution of synonymous codon usage in metazoans. *Current opinion in genetics & development*, 12(6), pp. 640–649.

Duret, L. (2008). Neutral theory: the null hypothesis of molecular evolution. *Nature Education*, 1, pp. 803–806.

Duret, L. and Mouchiroud, D. (1999). Expression pattern and, surprisingly, gene length shape codon usage in caenorhabditis, drosophila, and arabidopsis. *Proceedings of the National Academy of Sciences*, 96(8), pp. 4482–4487.

Erben, E. D. and Clayton, C. (2018). Codon usage in trypanosomatids: The bias of expression. *Trends in parasitology.*

Ferrer-Admetlla, A. et al. (2016). An approximate markov model for the wright–fisher diffusion and its application to time series data. *Genetics*, 203(2), pp. 831–846.

Fitzpatrick, D. A. et al. (2006). A fungal phylogeny based on 42 complete genomes derived from supertree and combined gene analysis. *BMC evolutionary biology*, 6(1), p. 99.

Fluitt, A., Pienaar, E. and Viljoen, H. (2007). Ribosome kinetics and aa-trna competition determine rate and fidelity of peptide synthesis. *Comput Biol Chem*, 31(5-6), pp. 335–46.

Fowlkes, E. B. and Mallows, C. L. (1983). A method for comparing two hierarchical clusterings. *Journal of the American statistical association*, 78(383), pp. 553–569.

Fox, J. M. and Erill, I. (2010). Relative codon adaptation: a generic codon bias index for prediction of gene expression. *DNA research*, 17(3), pp. 185–196.

Frumkin, I. et al. (2018). Codon usage of highly expressed genes affects proteome-wide translation efficiency. *Proceedings of the National Academy of Sciences*, 115(21), pp. E4940–E4949.

Fuglsang, A. (2005). On the methodological weakness of 'the effective number of codons': a reply to marashi and najafabadi. *Biochemical and biophysical research communications*, 327(1), pp. 1–3.

Ghaemmaghami, S. et al. (2003). Global analysis of protein expression in yeast. *Nature*, 425(6959), p. 737.

Gil, M. et al. (2013). Codonphyml: fast maximum likelihood phylogeny estimation under codon substitution models. *Molecular biology and evolution*, 30(6), pp. 1270–1280.

Gingold, H. and Pilpel, Y. (2011). Determinants of translation efficiency and accuracy. *Molecular systems biology*, 7(1), p. 481.

Gladitz, J. et al. (2005). Codon usage comparison of novel genes in clinical isolates of haemophilus influenzae. *Nucleic acids research*, 33(11), pp. 3644–3658.

Goffena, J. et al. (2018). Elongator and codon bias regulate protein levels in mammalian peripheral neurons. *Nature communications*, 9(1), p. 889.

Gouy, M. and Gautier, C. (1982). Codon usage in bacteria: correlation with gene expressivity. *Nucleic acids research*, 10(22), pp. 7055–7074.

Goz, E., Zafrir, Z. and Tuller, T. (2018). Universal evolutionary selection for high dimensional silent patterns of information hidden in the redundancy of viral genetic code. *Bioinformatics*.

Gribskov, M., Devereux, J. and Burgess, R. R. (1984). The codon preference plot: graphic analysis of protein coding sequences and prediction of gene expression.

Guo, X., Bao, J. and Fan, L. (2007). Evidence of selectively driven codon usage in rice: implications for gc content evolution of gramineae genes. *FEBS letters*, 581(5), pp. 1015–1021.

Hall, B. K. (2007). Homology and homoplasy. *Handbook of the philosophy of science Philosophy of biology*, pp. 429–453.

Hart, A. et al. (2018). Codon usage bias reveals genomic adaptations to environmental conditions in an acidophilic consortium. *PloS one*, 13(5), p. e0195869.

Hartl, D. L., Clark, A. G. and Clark, A. G. (1997). *Principles of population genetics*, vol. 116. Sinauer associates Sunderland, MA.

Hauber, D. J., Grogan, D. W. and DeBry, R. W. (2016). Mutations to less-preferred synonymous codons in a highly expressed gene of escherichia coli: fitness and epistatic interactions. *PloS one*, 11(1), p. e0146375.

Hershberg, R. and Petrov, D. A. (2008). Selection on codon bias. *Annual review of genetics*, 42, pp. 287–299.

Hibbett, D. S. et al. (2007). A higher-level phylogenetic classification of the fungi. *Mycological research*, 111(5), pp. 509–547.

Howe, K., Bateman, A. and Durbin, R. (2002). Quicktree: building huge neighbour-joining trees of protein sequences. *Bioinformatics*, 18(11), pp. 1546–1547.

Ikemura, T. (1981). Correlation between the abundance of escherichia coli transfer rnas and the occurrence of the respective codons in its protein genes. *Journal of molecular biology*, 146(1), pp. 1–21.

Ikemura, T. (1985). Codon usage and trna content in unicellular and multicellular organisms. *Molecular biology and evolution*, 2(1), pp. 13–34.

Jeacock, L., Faria, J. and Horn, D. (2018). Codon usage bias controls mrna and protein abundance in trypanosomatids. *Elife*, 7, p. e32496.

Johnson, A. P. et al. (2008). The miller volcanic spark discharge experiment. *Science*, 322(5900), pp. 404–404.

Jolliffe, I. T. and Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065), p. 20150202.

Karlin, S. and Mrázek, J. (2000). Predicted highly expressed genes of diverse prokaryotic genomes. *Journal of bacteriology*, 182(18), pp. 5238–5250.

Karlin, S., Mrázek, J. and Campbell, A. M. (1998). Codon usages in different gene classes of the escherichia coli genome. *Molecular microbiology*, 29(6), pp. 1341–1355.

Kaufmann, W. K. and Paules, R. S. (1996). Dna damage and cell cycle checkpoints. *The FASEB Journal*, 10(2), pp. 238–247.

Keller, I., Bensasson, D. and Nichols, R. A. (2007). Transition-transversion bias is not universal: a counter example from grasshopper pseudogenes. *PLoS genetics*, 3(2), p. e22.

Kimchi-Sarfaty, C. et al. (2007). A" silent" polymorphism in the mdr1 gene changes substrate specificity. *Science*, 315(5811), pp. 525–528.

Kimura, M. (1964). Diffusion models in population genetics. *Journal of Applied Probability*, 1(2), pp. 177–232.

Knight, R. D., Freeland, S. J. and Landweber, L. F. (2001). A simple model based on mutation and selection explains trends in codon and amino-acid usage and gc composition within and across genomes. *Genome biology*, 2(4), pp. research0010–1.

Koonin, E. V. (2005). Orthologs, paralogs, and evolutionary genomics. *Annu Rev Genet*, 39, pp. 309–338.

Koonin, E. V. and Galperin, M. (2013). *Sequence—evolution—function: computational approaches in comparative genomics.* Springer Science & Business Media.

Lange, K. (2003). Basic principles of population genetics. *Applied Probability*, pp. 1–20.

Lee, S. et al. (2010). Relative codon adaptation index, a sensitive measure of codon usage bias. *Evolutionary Bioinformatics*, 6, pp. EBO–S4608.

Liu, H. et al. (2017). Codon usage bias in 5 terminal coding sequences reveals distinct enrichment of gene functions. *Genomics*, 109(5-6), pp. 506–513.

Lobry, J. (1996). Asymmetric substitution patterns in the two dna strands of bacteria. *Molecular biology and evolution*, 13(5), pp. 660–665.

Lovmar, M. and Ehrenberg, M. (2006). Rate, accuracy and cost of ribosomes in bacterial cells. *Biochimie*, 88(8), pp. 951–961.

Lykke-Andersen, S. and Jensen, T. H. (2007). Overlapping pathways dictate termination of rna polymerase ii transcription. *Biochimie*, 89(10), pp. 1177–1182.

Maimon, O. and Rokach, L. (2009). Introduction to knowledge discovery and data mining. In *Data mining and knowledge discovery handbook*, Springer, pp. 1–15.

Marashi, S.-A. and Najafabadi, H. S. (2004). How reliable re-adjustment is: correspondence regarding a. fuglsang,"the 'effective number of codons' revisited". *Biochemical and biophysical research communications*, 324(1), pp. 1–2.

Mazumder, T. H., Chakraborty, S. and Paul, P. (2014). A cross talk between codon usage bias in human oncogenes. *Bioinformation*, 10(5), pp. 256–62.

McLachlan, A. D., Staden, R. and Boswell, D. R. (1984). A method for measuring the non-random bias of a codon usage table. *Nucleic acids research*, 12(24), pp. 9567–9575.

Mondal, S. K. et al. (2016). Analysis of phylogeny and codon usage bias and relationship of gc content, amino acid composition with expression of the structural nif genes. *Journal of Biomolecular Structure and Dynamics*, 34(8), pp. 1649–1666.

Moran, P. A. P. (1958). Random processes in genetics. In *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 54, Cambridge University Press, pp. 60–71.

Moriyama, E. N. and Powell, J. R. (1998). Gene length and codon usage bias in drosophila melanogaster, saccharomyces cerevisiae and escherichia coli. *Nucleic acids research*, 26(13), pp. 3188–3193.

Muto, A. and Osawa, S. (1987). The guanine and cytosine content of genomic dna and bacterial evolution. *Proceedings of the National Academy of Sciences*, 84(1), pp. 166–169.

Nabiyouni, M., Prakash, A. and Fedorov, A. (2013). Vertebrate codon bias indicates a highly gc-rich ancestral genome. *Gene*, 519(1), pp. 113–119.

Nakahigashi, K. et al. (2014). Effect of codon adaptation on codon-level and gene-level translation efficiency in vivo. *BMC Genomics*, 15, p. 1115.

Neafsey, D. E. and Galagan, J. E. (2007). Positive selection for unpreferred codon usage in eukaryotic genomes. *BMC evolutionary biology*, 7(1), p. 119.

Nirenberg, M. et al. (1965). Rna codewords and protein synthesis, vii. on the general nature of the rna code. *Proceedings of the National Academy of Sciences*, 53(5), pp. 1161–1168.

O'Donnell, S. M. and Janssen, G. R. (2001). The initiation codon affects ribosome binding and translational efficiency in escherichia coli of ci mrna with or without the 5 untranslated leader. *Journal of bacteriology*, 183(4), pp. 1277–1283.

Parker, S. et al. (2018). Large-scale profiling of noncoding rna function in yeast. *PLoS genetics*, 14(3), p. e1007253.

Parmley, J. L., Chamary, J. and Hurst, L. D. (2005). Evidence for purifying selection against synonymous mutations in mammalian exonic splicing enhancers. *Molecular biology and evolution*, 23(2), pp. 301–309.

Pearson, W. R. and Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences*, 85(8), pp. 2444–2448.

Percudani, R., Pavesi, A. and Ottonello, S. (1997). Transfer rna gene redundancy and translational selection in saccharomyces cerevisiae. *Journal of molecular biology*, 268(2), pp. 322–330.

Petrov, D. A. (2002). Mutational equilibrium model of genome size evolution. *Theoretical population biology*, 61(4), pp. 531–544.

Pintó, R. M. et al. (2018). Hepatitis a virus codon usage: Implications for translation kinetics and capsid folding. *Cold Spring Harbor perspectives in medicine*, p. a031781.

Polani, D. (2002). Measures for the organization of self-organizing maps. In *Self-Organizing neural networks*, Springer, pp. 13–44.

Pouyet, F. et al. (2016). Senca: a multilayered codon model to study the origins and dynamics of codon usage. *Genome biology and evolution*, 8(8), pp. 2427–2441.

Puigbò, P., Bravo, I. G. and Garcia-Vallve, S. (2008). Caical: a combined set of tools to assess codon usage adaptation. *Biology direct*, 3(1), p. 38.

Qin, H. et al. (2004). Intragenic spatial patterns of codon usage bias in prokaryotic and eukaryotic genomes. *Genetics*, 168(4), pp. 2245–2260.

Reis, M. d., Savva, R. and Wernisch, L. (2004). Solving the riddle of codon usage preferences: a test for translational selection. *Nucleic acids research*, 32(17), pp. 5036–5044.

Roy, B. et al. (2015). Nonsense suppression by near-cognate trnas employs alternative base pairing at codon positions 1 and 3. *Proc Natl Acad Sci U S A*, 112(10), pp. 3038–43.

Roymondal, U., Das, S. and Sahoo, S. (2009). Predicting gene expression level from relative codon usage bias: an application to escherichia coli genome. *DNA research*, 16(1), pp. 13–30.

Santos, M. A. and Tuite, M. F. (1995). The cug codon is decoded in vivo as serine and not leucine in candida albicans. *Nucleic acids research*, 23(9), pp. 1481–1486.

Savisaar, R. and Hurst, L. D. (2018). Exonic splice regulation imposes strong selection at synonymous sites. *Genome research*, 28(10), pp. 1442–1454.

Sesma, A. and Von der Haar, T. (2014). *Fungal RNA Biology*. Springer.

Sharp, P. M. and Li, W.-H. (1986). Codon usage in regulatory genes in escherichia coli does not reflect selection for 'rare'codons. *Nucleic acids research*, 14(19), pp. 7737–7749.

Sharp, P. M. and Li, W.-H. (1987). The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic acids research*, 15(3), pp. 1281–1295.

Shields, D. C. et al. (1988). " silent" sites in drosophila genes are not neutral: evidence of selection among synonymous codons. *Molecular biology and evolution*, 5(6), pp. 704–716.

Shu, J.-J. (2017). A new integrated symmetrical table for genetic codes. *Biosystems*, 151, pp. 21–26.

Song, H. et al. (2017). Comprehensive analysis of correlations among codon usage bias, gene expression, and substitution rate in arachis duranensis and arachis ipaënsis orthologs. *Scientific reports*, 7(1), p. 14853.

Stearns, S. C. and Hoekstra, R. F. (2000). *Evolution, an introduction*. Oxford University Press.

Stergachis, A. B. et al. (2013). Exonic transcription factor binding directs codon choice and affects protein evolution. *Science*, 342(6164), pp. 1367–1372.

Stoltzfus, A. and Norris, R. W. (2015). On the causes of evolutionary transition: transversion bias. *Molecular biology and evolution*, 33(3), pp. 595–602.

Sueoka, N. (1988). Directional mutation pressure and neutral molecular evolution. *Proceedings of the National Academy of Sciences*, 85(8), pp. 2653–2657.

Suzuki, H., Saito, R. and Tomita, M. (2004). The 'weighted sum of relative entropy': a new index for synonymous codon usage bias. *Gene*, 335, pp. 19–23.

Taylor, F. and Coates, D. (1989). The code within the codons. *Biosystems*, 22(3), pp. 177–187.

Thommen, M., Holtkamp, W. and Rodnina, M. V. (2017). Co-translational protein folding: progress and methods. *Current opinion in structural biology*, 42, pp. 83–89.

Thompson, R. C. and Karim, A. M. (1982). The accuracy of protein biosynthesis is limited by its speed: high fidelity selection by ribosomes of aminoacyl-trna ternary complexes containing gtp [gamma s]. *Proceedings of the National Academy of Sciences*, 79(16), pp. 4922–4926.

Tian, J. P. (2007). *Evolution algebras and their applications*. Springer.

Tlusty, T. (2007). A model for the emergence of the genetic code as a transition in a noisy information channel. *Journal of theoretical biology*, 249(2), pp. 331–342.

Tlusty, T. (2008). Rate-distortion scenario for the emergence and evolution of noisy molecular codes. *Physical review letters*, 100(4), p. 048101.

Tuller, T. et al. (2010). An evolutionarily conserved mechanism for controlling the efficiency of protein translation. *Cell*, 141(2), pp. 344–354.

Uddin, A. (2017). Indices of codon usage bias. *Proteom Bioinform*, 10(6).

Uddin, A. and Chakraborty, S. (2018). Codon usage pattern of genes involved in central nervous system. *Molecular Neurobiology*, pp. 1–12.

Urrutia, A. O. and Hurst, L. D. (2001). Codon usage bias covaries with expression breadth and the rate of synonymous evolution in humans, but this is not evidence for selection. *Genetics*, 159(3), pp. 1191–1199.

Uzman, A. (2003). Molecular biology of the cell: Alberts, b., johnson, a., lewis, j., raff, m., roberts, k., and walter, p.

Wagner, S. and Wagner, D. (2007). *Comparing clusterings: an overview.* Universität Karlsruhe, Fakultät für Informatik Karlsruhe.

Wan, X.-F. et al. (2004). Quantitative relationship between synonymous codon usage bias and gc composition across unicellular genomes. *BMC Evolutionary Biology*, 4(1), p. 19.

Wang, M. et al. (2012). Paxdb, a database of protein abundance averages across all three domains of life. *Molecular & cellular proteomics*, 11(8), pp. 492–500.

Wang, Q., Parrish, A. R. and Wang, L. (2009). Expanding the genetic code for biological studies. *Chemistry & biology*, 16(3), pp. 323–336.

Watanabe, K. and Osawa, S. (1995). trna sequences and variations in the genetic code. *D S [tilde] oll and UL RajBhandary (ed), tRNA: structure, biosynthesis, and function American Society for Microbiology, Washington, DC*, pp. 225–250.

West, G. B., Brown, J. H. and Enquist, B. J. (2000). The origin of universal scaling laws in biology. *Scaling in biology*, pp. 87–112.

Whitehead, A. and Crawford, D. L. (2006). Variation within and among species in gene expression: raw material for evolution. *Molecular ecology*, 15(5), pp. 1197–1211.

Woese, C. R., Kandler, O. and Wheelis, M. L. (1990). Towards a natural system of organisms: proposal for the domains archaea, bacteria, and eucarya. *Proceedings of the National Academy of Sciences*, 87(12), pp. 4576–4579.

Wong, J. T.-F. (1975). A co-evolution theory of the genetic code. *Proceedings of the National Academy of Sciences of the United States of America*, 72(5), p. 1909.

Wright, F. (1990). The 'effective number of codons' used in a gene. *Gene*, 87(1), pp. 23–29.

Xia, X. (1996). Maximizing transcription efficiency causes codon usage bias. *Genetics*, 144(3), pp. 1309–1320.

Xia, X. (2007). An improved implementation of codon adaptation index. *Evolutionary Bioinformatics*, 3, p. 117693430700300028.

Yakovchuk, P., Protozanova, E. and Frank-Kamenetskii, M. D. (2006). Base-stacking and base-pairing contributions into thermal stability of the dna double helix. *Nucleic acids research*, 34(2), pp. 564–574.

Yang, Z. and Nielsen, R. (2008). Mutation-selection models of codon substitution and their use to estimate selective strengths on codon usage. *Molecular biology and evolution*, 25(3), pp. 568–579.

Yannai, A., Katz, S. and Hershberg, R. (2018). The codon usage of lowly expressed genes is subject to natural selection. *Genome biology and evolution*, 10(5), pp. 1237–1246.

Zeng, K. and Charlesworth, B. (2009). Estimating selection intensity on synonymous codon usage in a nonequilibrium population. *Genetics*, 183(2), pp. 651–662.

Zoller, S. and Schneider, A. (2012). A new semiempirical codon substitution model based on principal component analysis of mammalian sequences. *Mol Biol Evol*, 29(1), pp. 271–7.

Zörnig, P. and Altmann, G. (1995). Unified representation of zipf distributions. *Computational Statistics & Data Analysis*, 19(4), pp. 461–473.

# Appendices

# Appendix A

# Major Programs for This Work

All the datasets are available online[1]. Detailed structure of the data files is explained in ReadMe.txt.

Programs required to achieve the above databases are summarised in Table 22.

[1] https://www.cs.kent.ac.uk/projects/statthermcub/

Table 22: Main programs list and relevant function explanation

| program | function |
|---|---|
| kingdomDownload | create downloading list from Ensemble database |
| chopSeq | python script to preprocess genome files |
| getCodonSequence | Read genes from genome files transformed them into codon sequences |
| getGeneName | Read gene names from genome files |
| geneSynoRatio | codon occurrence configurations in each gene |
| synoTable2 | global codon usage table for all the species |
| SetSynonymouCodonTable | set underlying codon usage probability equal or biased |
| *AminoAcidH | 18 programs for 18 amino acids individually obtain multinomial distribution probability |
| EforMore | find maximum multinomial distribution probability for a certain length |
| ForT* | 4 types of datasets for further analysis T,Ta,Tb,Tab refers to Table 21 |
| getHomoInfor | retrieve homology information online |
| clusterComp | similarity quantification between cluster trees |

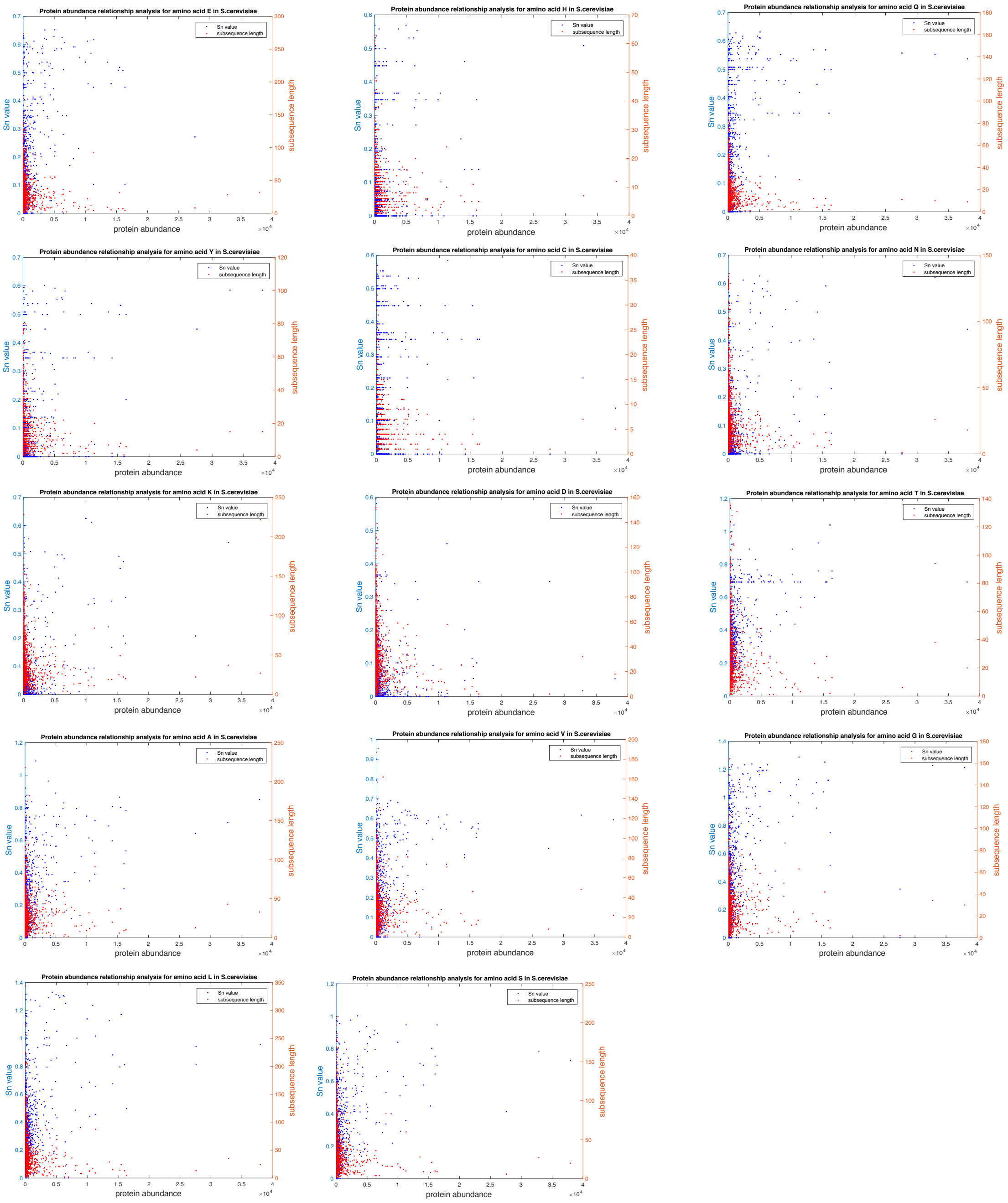
# Appendix B

# Supplement Figures to the Main Contents

## B.1 Relationships among Protein Abundance, $Sn$ and subsequence Length in *S.cerevisiae*

## B.2 Cooperation between $Sn$ and Gene Length for Protein Production

## B.3 $Sn$ Distribution for Different Amino Acids in *S.cerevisiae*



Protein abundance relationship analysis for amino acid E in S.cerevisiae

Protein abundance relationship analysis for amino acid H in S.cerevisiae

Protein abundance relationship analysis for amino acid Q in S.cerevisiae

Protein abundance relationship analysis for amino acid Y in S.cerevisiae

Protein abundance relationship analysis for amino acid C in S.cerevisiae

Protein abundance relationship analysis for amino acid N in S.cerevisiae

Protein abundance relationship analysis for amino acid K in S.cerevisiae

Protein abundance relationship analysis for amino acid D in S.cerevisiae

Protein abundance relationship analysis for amino acid T in S.cerevisiae

Protein abundance relationship analysis for amino acid A in S.cerevisiae

Protein abundance relationship analysis for amino acid V in S.cerevisiae

Protein abundance relationship analysis for amino acid G in S.cerevisiae

Protein abundance relationship analysis for amino acid L in S.cerevisiae

Protein abundance relationship analysis for amino acid S in S.cerevisiae

Figure 32: *Sn* and subsequence length against protein abundance: supplement to Figure 7

.

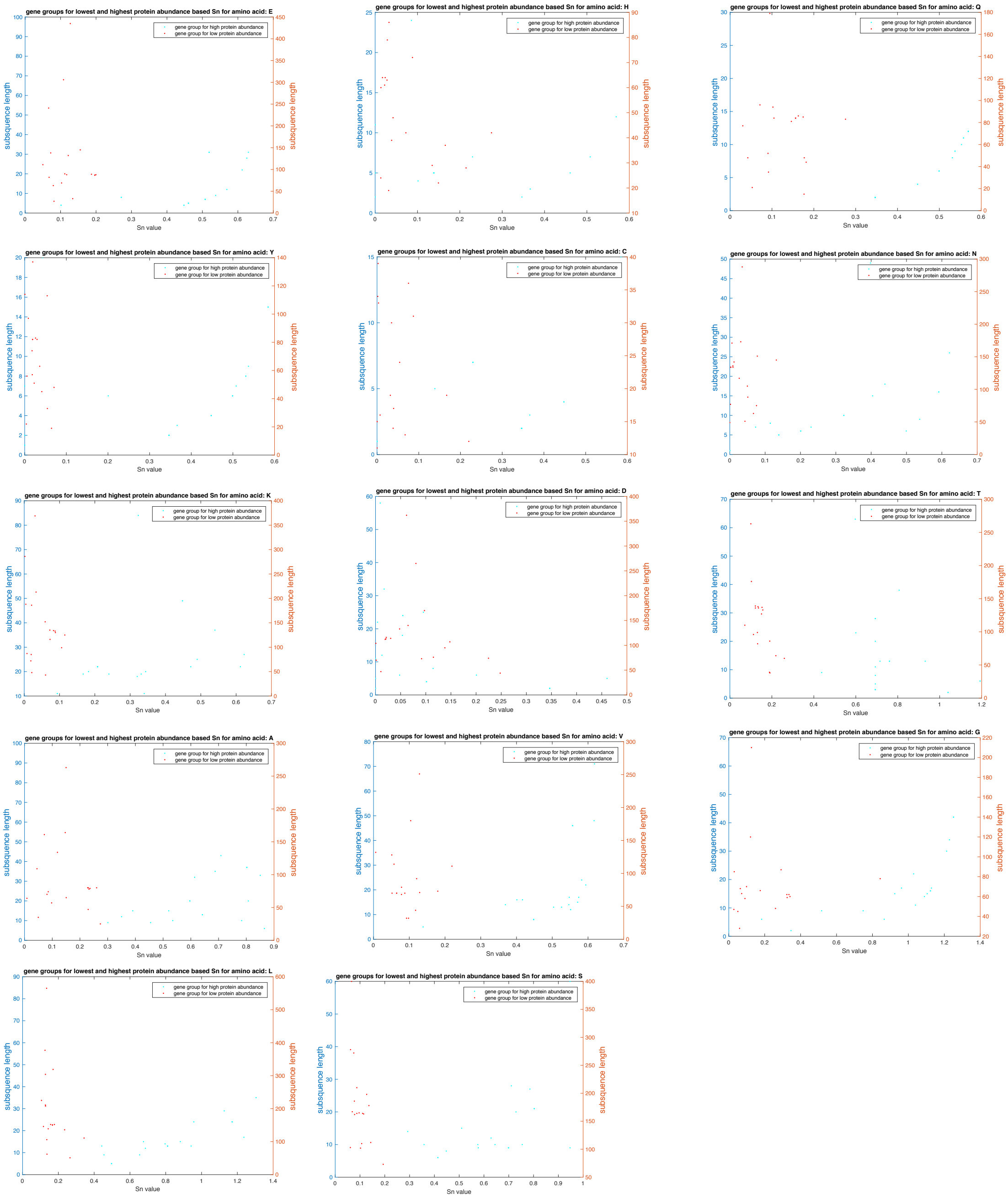


Figure 33: $Sn$ against subsequence length in two groups, supplement to Figure 9.

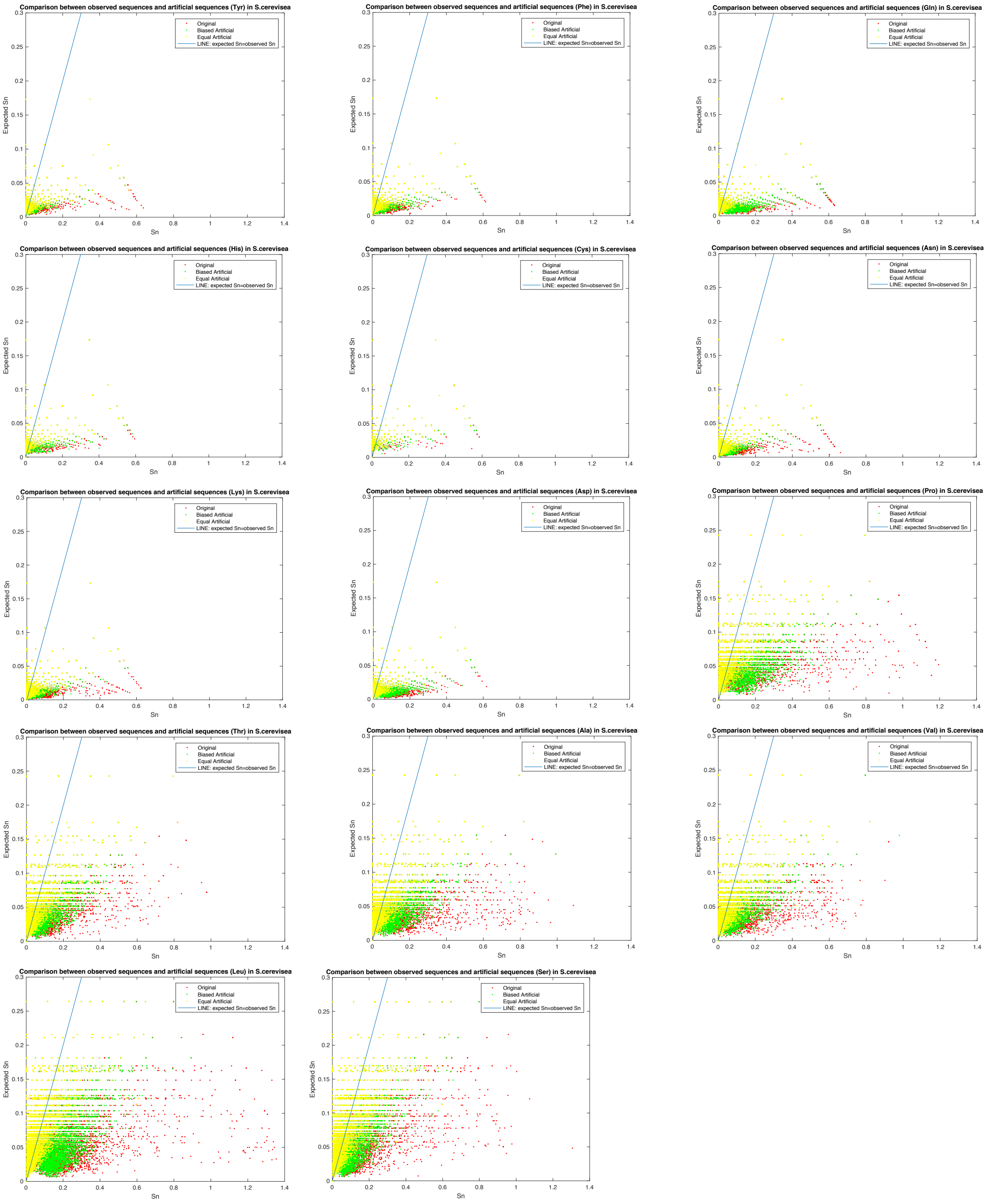
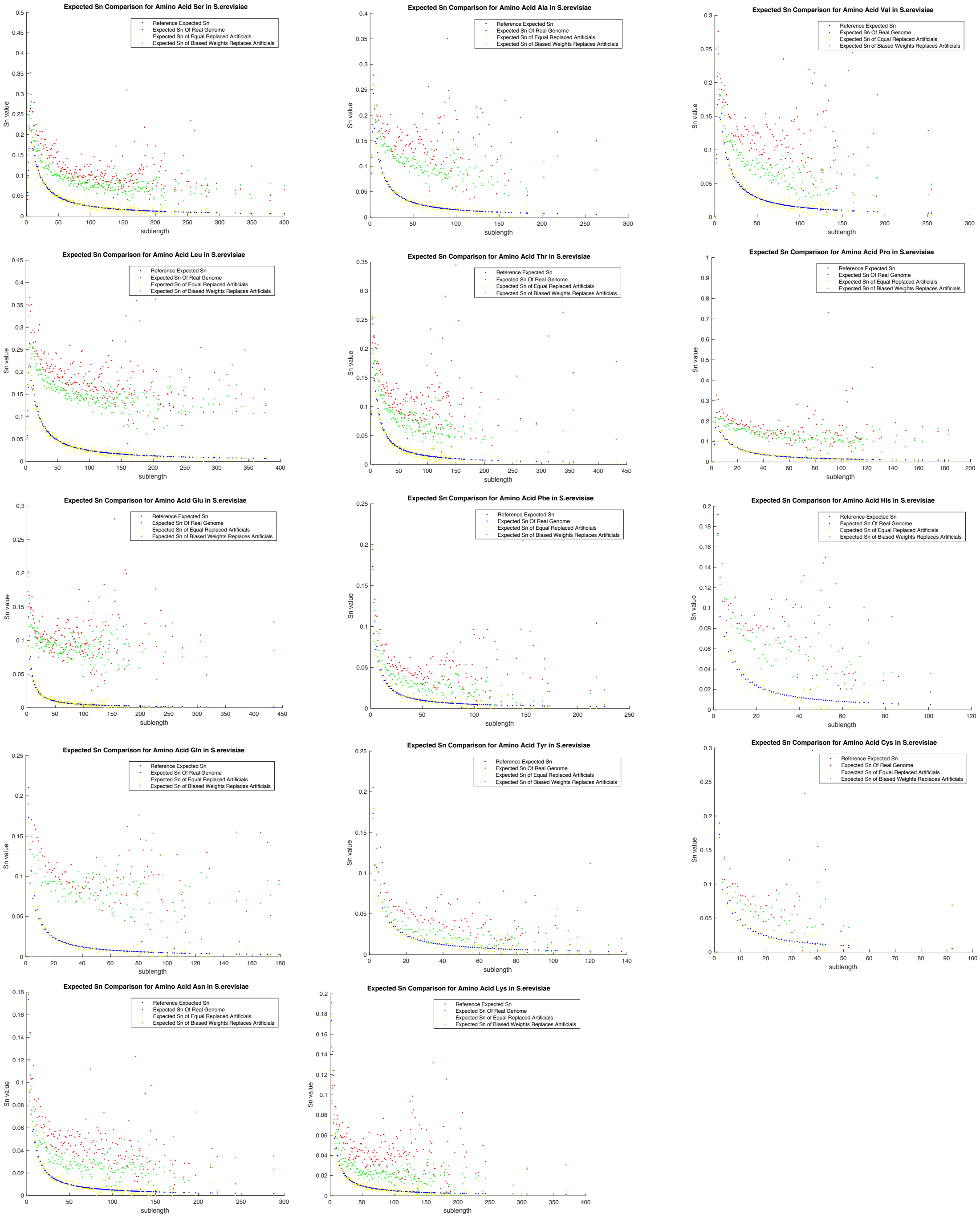


Figure 34: $Sn$ distribution overview: supplements to Figure12

.



Expected Sn Comparison for Amino Acid Ser in S.erevisiae

Expected Sn Comparison for Amino Acid Ala in S.erevisiae

Expected Sn Comparison for Amino Acid Val in S.erevisiae

Expected Sn Comparison for Amino Acid Leu in S.erevisiae

Expected Sn Comparison for Amino Acid Thr in S.erevisiae

Expected Sn Comparison for Amino Acid Pro in S.erevisiae

Expected Sn Comparison for Amino Acid Glu in S.erevisiae

Expected Sn Comparison for Amino Acid Phe in S.erevisiae

Expected Sn Comparison for Amino Acid His in S.erevisiae

Expected Sn Comparison for Amino Acid Gln in S.erevisiae

Expected Sn Comparison for Amino Acid Tyr in S.erevisiae

Expected Sn Comparison for Amino Acid Cys in S.erevisiae

Expected Sn Comparison for Amino Acid Asn in S.erevisiae

Expected Sn Comparison for Amino Acid Lys in S.erevisiae

Reference Expected Sn
Expected Sn Of Real Genome
Expected Sn of Equal Replaced Artificials
Expected Sn of Biased Weights Replaces Artificials
Sn value
sublength

Figure 35: $Sn$ distribution against length: supplement to Figure14

.

# Appendix C

# Byproducts

## C.1 Methods to Combine $Sn$ into a Genome-wide Measure

We made a lot of efforts to find a reasonable and feasible way to combine $Sn$ into a genome wide measure, among which $KL_{value}$ is a good choice but finally we did not adopt it when taking into consider of the computing time (10 times slowly than $MD$ measure). Here we briefly display our results using this approach, which is time cost but with high accuracy.

KL divergence (short for 'KL' value) is a measure of similarity between two probability distributions, shown as Equation 35.

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \tag{35}$$

CUB of the whole genome $KL_{value}$: in per genome P is the distribution of observed Sn values of genes, while Q is the theoretical distribution which is obtained as follows: Supposing in the interested genome, there are N sequences, the $i$-th sequence is length $L_i$ (i $\in$ [ 1, N] ). For the certain length $L_i$, all the possible configurations constitute reference $Sn$ distribution annotated as '$DistributionO_i$'. Thus for all the N sequences, we obtain N distributions of '$DistributionsO_i$'. Normalise all the N distributions to the same scale by standardise bin size and adding '0' of Sn values to short lengths ($Sn$ value should be x axis in plot), and then

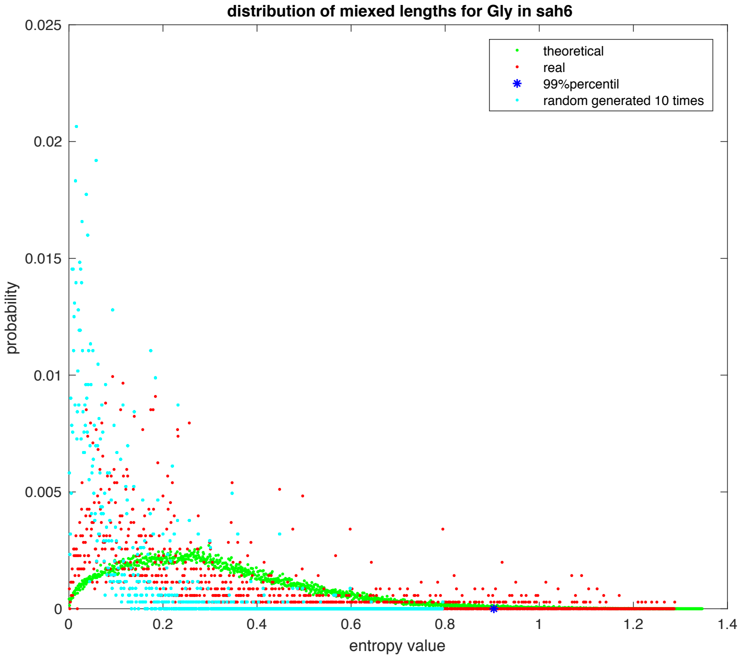


Figure 36: $KL_{value}$ Method explanation: example of CUB for amino acid Gly in species $S.arboricola$; along x axis are $Sn$ values: green curve is the $Sn$ distribution reference Q, y axis is the probability of such $Sn$ value in the whole genome. Final $KL_{value}$ of Gly in $S.arboricola$ is calculated based on KL divergence between observed $Sn$ distribution P and reference $Sn$ distribution Q.

add up all the probability density values in each bin among N distributions, finally normalising the unit summations by dividing N, then the final theoretical distribution Q is achieved.

An example result based on KL divergence is shows in Figure 35.

## C.2 Hierarchical Clustering Based on $Sn$

### C.2.1 Comparisons Between CUB Cluster Trees Between Species

We perform hierarchical clustering analysis for 18 amino acids within one genome based on $Sn$ datasets. Analysis of 6 species with certain phylogenetic relationships are shown in Figure 37. Visually the shape of CUB cluster trees at the amino acid level seem to resemble according to phylogenetically close related species, further we verified such similarity with quantification by Fowlks method.

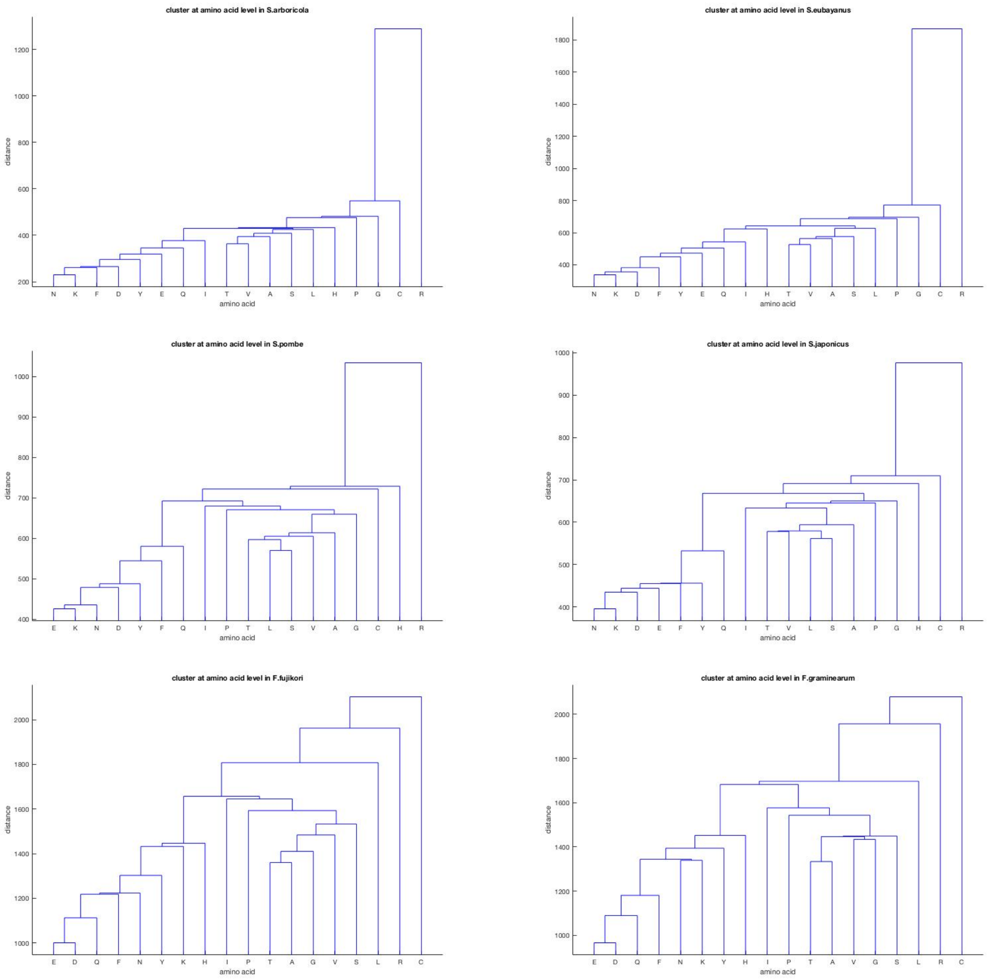


Figure 37: CUB cluster trees of 18 amino acids in 6 species *S.arboricola*, *S.eubayanus*, *S.pombe*, *S.japonicus*, *F.fujikori*, *F.graminearum*. X axis represents amino acids and Y axis shows cluster distances. Meanwhile inspecting Phylogenetic relationships refer to Table 15, we visually spot that similar cluster structures tend to exist between phylogenetically intimate species. This suggests that codon usage bias of amino acids tends to correlate with species phylogenetic taxonomy.

The results of similarity quantification between species are shown in Table 23, from which we see CUB clusters of *S.arboricola* and *S.eubayanus* have similarity of high significance at confidence level $\alpha$=0.05, and the same case is as species pair *F.fujikori* and *F.graminearum*. Further more, species pairs *S.pombe* and *S.japonicus* have a smaller similarity value compared to other two pairs. Referring to phylogenetic tree as Figure 19, it is conveyed that if species are more closely

Table 23: Clusters Similarity Quantification of Codon Usage Bias between Species

| Species Pair | Similarity | Dissimilarity Confidence Interval |
|---|---|---|
| *S.arboricola* and *S.eubayanus* | 1 | 0.3073, 0.7123 |
| *S.arboricola* and *F.fujikuroi* | 0.5501 | 0.2421,0.6371 |
| *F.fujikuroi* and *F.graminearum* | 1 | 0.1846,0.5736 |
| *S.arboricola* and *S.pombe* | 0.7051 | 0.3073, 0.7123 |
| *S.pombe* and *S.japonicus* | 0.8462 | 0.3073, 0.7123 |

related in phylogenetic terms, the relationship between their amino-acid specific codon usage patterns is more similar than if species are less closely related, which agrees with our SOM analysis as stated in section 3.3.1.

In 1972 King and Hare suggested that amino acid composition could be similarly used as a taxonomic character. Our CUB analysis supports this suggestion in a way that: the relationship similarity of CUB among amino acids are related to the phylogenetic relationships between these species and that amino acid composition could indeed play a role as a taxonomic character in the study of phylogenetic relationships.

### C.2.2 Comparison Between CUB Cluster Trees and Cluster Tree Derived from Amino Acid Properties

Further more, we explore correlation between CUB and amino acid physical properties. Hierarchical cluster trees of physical properties of amino acids in species *S.arboricola* are shown as Figure 38. By way of comparing amino acid physical property hierarchical cluster tree and codon usage bias hierarchical cluster tree in interested species, we are able to discover which amino acid physical properties may impact codon usage evolution.

When we perform cluster similarity quantification between codon usage bias and amino acid physical properties, an example species *Sporisorium Reilianum* shows result as Table 24. In this table, $B_k$ values in rows of Hydrophobicity Index and Conservation Index are located outside the corresponding dissimilarity confidence intervals, and hence we judge that there exists significant similarity between clusters of CUB and clusters of such amino acid physical properties. To be specific, at the statistical significance level $\alpha$=0.05, CUB of species *Sporisorium Reilianum* correlate with amino acid Hydrophobicity Index and Conservation Index.

Table 24: Clusters Similarity between CUB and Amino Acids Physical Properties in species *Sporisorium Reilianum*

| Physical Properties | Similarity | Dissimilarity Confidence Interval ($\alpha$=0.05) |
|---|---|---|
| Molecular_Weight | 0.3615 | 0.1669, 0.4839 |
| NH2_pKA | 0.3326 | 0.1793, 0.4889 |
| COOH_pKA | 0.2611 | 0.1866, 0.4642 |
| Side_Chain_pKA | 0.2292 | 0.1686, 0.5727 |
| pI | 0.2751 | 0.1970, 0.6108 |
| No._Atoms | 0.3176 | 0.1705, 0.3645 |
| Volume | 0.3491 | 0.1600, 0.3641 |
| Hydrophobicity_Index | 0.4537 | 0.1652, 0.4110 |
| Conservation_Index | 0.4608 | 0.1552, 0.4406 |
| rel_C_cost | 0.2269 | 0.1652, 0.4110 |
| rel_N_cost | 0.3963 | 0.1982, 0.5274 |
| rel_S_cost | 0.3873 | 0.2675, 0.6775 |
| rel_glucose | 0.4819 | 0.1382, 0.4857 |
| Synthesis_Steps | 0.3511 | 0.1822, 0.4136 |

We select disease related fungal species to perform the same analysis as *Sporisorium Reilianum*, aiming to find correlation between CUB and amino acid properties in those species. We refer to KEGG database to retrieve information about disease related fungal species. Results are shown in Table 38.

For disease related fungal species, the results of tree comparisons reveal interesting information: normally fungal pathogen infect skin and lung, and the species infect neither skin nor lung such as nervous system, CUB values are correlated to amino acid volume (molecular weight) and relative glucose cost, however not correlated to hydrophobicity index which shows correlation in skin and lung infectious species. For each species, 14 impact factors are investigated which are Molecular weight, NH2-pKA, COOH−pKA, Side_Chain_pKA, PI, No.of Atoms, Volume, Hydrophobicity_index, Conservation_index, rel_c_cost, rel_N_cost, rel_S_cost, rel_glucose, and synthesis_steps.

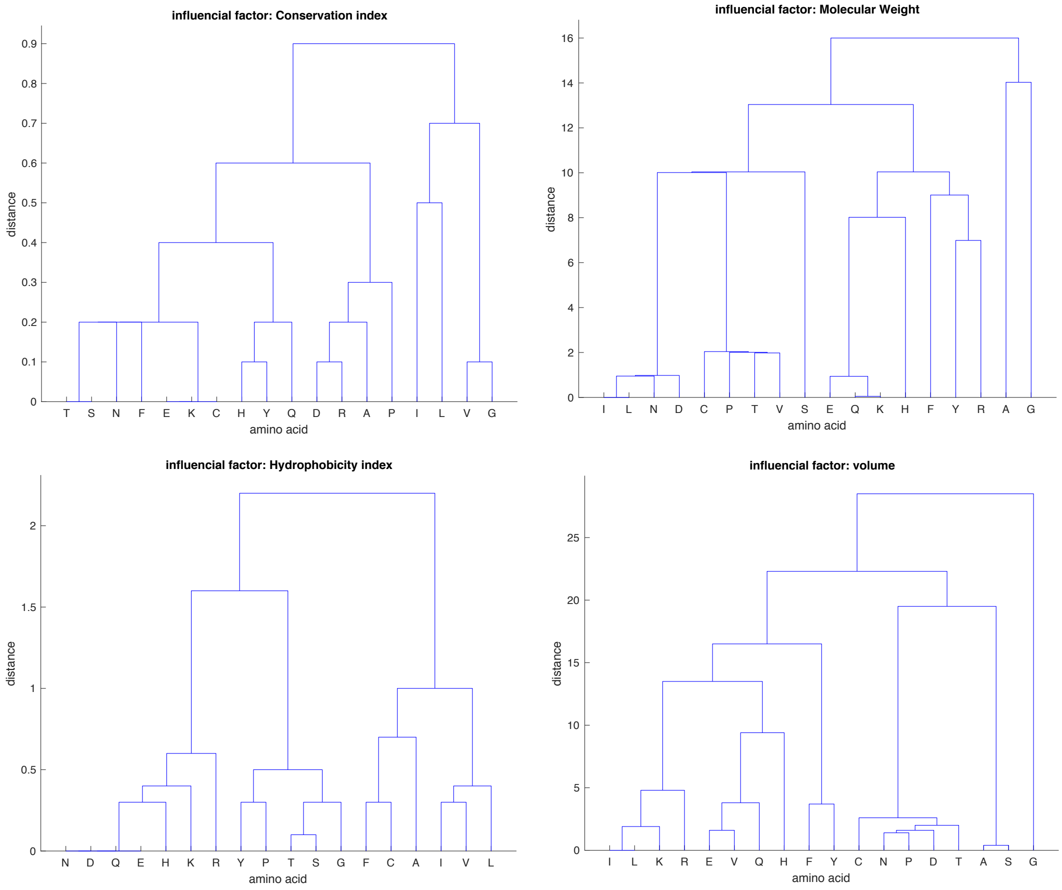
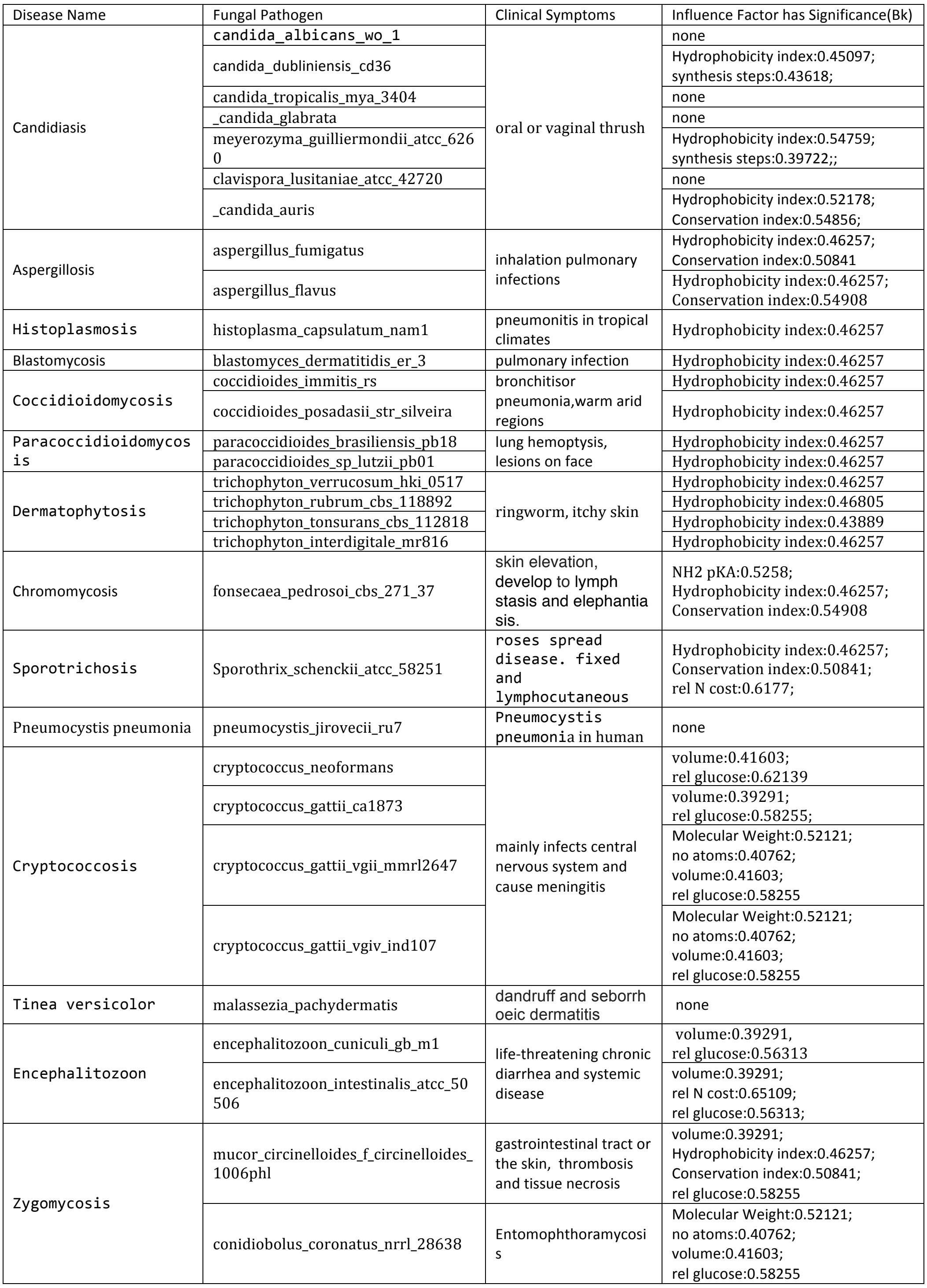


Figure 38: We obtain hierarchical cluster trees exploring amino acids physical properties in species *S.arboricola*, which include Molecular Weight, NH2_pKA, COOH_pKA, side_chain_pKA , pI, Number of atoms, volume, Hydrophobicity_index, Conservation_index, rel_C_cost, rel_N_cost, rel_S_cost, rel_glucose, synthesis_steps. And here we display hierarchical cluster trees of conservation index, molecular weight, hydrophobicity index and volume as an example.



| Disease Name | Fungal Pathogen | Clinical Symptoms | Influence Factor has Significance(Bk) |
|---|---|---|---|
| Candidiasis | candida_albicans_wo_1 | oral or vaginal thrush | none |
| | candida_dubliniensis_cd36 | | Hydrophobicity index:0.45097;<br>synthesis steps:0.43618; |
| | candida_tropicalis_mya_3404 | | none |
| | _candida_glabrata | | none |
| | meyerozyma_guilliermondii_atcc_6260 | | Hydrophobicity index:0.54759;<br>synthesis steps:0.39722;; |
| | clavispora_lusitaniae_atcc_42720 | | none |
| | _candida_auris | | Hydrophobicity index:0.52178;<br>Conservation index:0.54856; |
| Aspergillosis | aspergillus_fumigatus | inhalation pulmonary infections | Hydrophobicity index:0.46257;<br>Conservation index:0.50841 |
| | aspergillus_flavus | | Hydrophobicity index:0.46257;<br>Conservation index:0.54908 |
| Histoplasmosis | histoplasma_capsulatum_nam1 | pneumonitis in tropical climates | Hydrophobicity index:0.46257 |
| Blastomycosis | blastomyces_dermatitidis_er_3 | pulmonary infection | Hydrophobicity index:0.46257 |
| Coccidioidomycosis | coccidioides_immitis_rs | bronchitisor pneumonia,warm arid regions | Hydrophobicity index:0.46257 |
| | coccidioides_posadasii_str_silveira | | Hydrophobicity index:0.46257 |
| Paracoccidioidomycosis | paracoccidioides_brasiliensis_pb18 | lung hemoptysis, lesions on face | Hydrophobicity index:0.46257 |
| | paracoccidioides_sp_lutzii_pb01 | | Hydrophobicity index:0.46257 |
| Dermatophytosis | trichophyton_verrucosum_hki_0517 | ringworm, itchy skin | Hydrophobicity index:0.46257 |
| | trichophyton_rubrum_cbs_118892 | | Hydrophobicity index:0.46805 |
| | trichophyton_tonsurans_cbs_112818 | | Hydrophobicity index:0.43889 |
| | trichophyton_interdigitale_mr816 | | Hydrophobicity index:0.46257 |
| Chromomycosis | fonsecaea_pedrosoi_cbs_271_37 | skin elevation, develop to lymph stasis and elephantia sis. | NH2 pKA:0.5258;<br>Hydrophobicity index:0.46257;<br>Conservation index:0.54908 |
| Sporotrichosis | Sporothrix_schenckii_atcc_58251 | roses spread disease. fixed and lymphocutaneous | Hydrophobicity index:0.46257;<br>Conservation index:0.50841;<br>rel N cost:0.6177; |
| Pneumocystis pneumonia | pneumocystis_jirovecii_ru7 | Pneumocystis pneumonia in human | none |
| Cryptococcosis | cryptococcus_neoformans | mainly infects central nervous system and cause meningitis | volume:0.41603;<br>rel glucose:0.62139 |
| | cryptococcus_gattii_ca1873 | | volume:0.39291;<br>rel glucose:0.58255; |
| | cryptococcus_gattii_vgii_mmrl2647 | | Molecular Weight:0.52121;<br>no atoms:0.40762;<br>volume:0.41603;<br>rel glucose:0.58255 |
| | cryptococcus_gattii_vgiv_ind107 | | Molecular Weight:0.52121;<br>no atoms:0.40762;<br>volume:0.41603;<br>rel glucose:0.58255 |
| Tinea versicolor | malassezia_pachydermatis | dandruff and seborrh oeic dermatitis | none |
| Encephalitozoon | encephalitozoon_cuniculi_gb_m1 | life-threatening chronic diarrhea and systemic disease | volume:0.39291,<br>rel glucose:0.56313 |
| | encephalitozoon_intestinalis_atcc_50506 | | volume:0.39291;<br>rel N cost:0.65109;<br>rel glucose:0.56313; |
| Zygomycosis | mucor_circinelloides_f_circinelloides_1006phl | gastrointestinal tract or the skin, thrombosis and tissue necrosis | volume:0.39291;<br>Hydrophobicity index:0.46257;<br>Conservation index:0.50841;<br>rel glucose:0.58255 |
| | conidiobolus_coronatus_nrrl_28638 | Entomophthoramycosis | Molecular Weight:0.52121;<br>no atoms:0.40762;<br>volume:0.41603;<br>rel glucose:0.58255 |

Figure 39: Fungal pathogen study