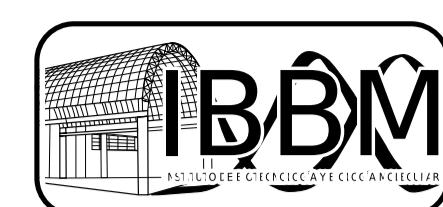


CUBES and CUBACR, script packages for the analysis of evolutionary traits of codon bias.



Mauricio Javier Lozano*, José Luis López, María Laura Fabre, and Antonio Lagares



IBBM - Instituto de Biotecnología y Biología Molecular, CONICET, CCT-La Plata,
Departamento de Ciencias Biológicas, Facultad de Ciencias Exactas, Universidad Nacional de
La Plata, calles 47 y 115, 1900-La Plata, Argentina..



UNIVERSIDAD NACIONAL
DE LA PLATA

Facultad de
Ciencias Exactas



Download a copy!

BACKGROUND

The balance between mutational biases and natural selection generates a wide range of GC contents and codon usage biases in Prokaryote genomes. Synonymous codons are selected to optimize translation of genes with different functions and expression levels, and to preserve the fitness of the cell. The choice of such optimal codons produce intragenomic codon usage heterogeneities³. The analysis of core-gene sets with increasing ancestries in bacteria, revealed an increased degree of adaptation of the most ancestral genes to the translational machinery⁸. This adaptation could be a consequence of codon selection for better translation accuracy or efficiency. One way to attempt differentiating those effects is to compare the codon usage of conserved vs. non-conserved genetic regions. Here we present bioinformatic tools which can perform several codon usage analysis on sets of genes belonging to different core-genomes.

What is a core-genome?

A core-genome is a set of genes which present an orthologous in all (or a high proportion) of the genomes in a given collection —i.e. all the common genes. Adding more genomes to the collection will reduce the core-genome, generally maintaining the genes which are related to fundamental cellular functions, and that were most likely present in the common ancestor. Using software such as EDGAR² or Get_Homologues⁴, it is possible to search for all the orthologous genes between any number of genomes. Opposite to core-genomes, singletons are genes which are only present in one (or a low proportion) of the genomes under study, and are usually more related to newly acquired accessory functions.

What is the codon usage Bias?

Proteins are polymers of 20 different types of amino acids, each one encoded in the RNA which in turn is transcribed from the DNA by a codon (a sequence of 3 nucleotidic bases). Since there are 4 types of nucleotides (A,T,C,G), a total of 64 codons exist. Of these, 3 codons (TAA, TTA, TGA) are used as translation stop signals, while the remaining 61 are used to encode for the 20 amino acids. The fact that there are more codons than amino acids makes the genetic code degenerate. Some amino acids are encoded by 1 codon only, while others are encoded by 2, 3, 4 or 6 codons. However, all the codons encoding the same amino acid -codon usage- are not used with the same frequency. This generates a codon usage bias which has been associated to gene-expression level, the adaptation of the codon usage to the proportion of transfer RNAs (tRNAs) of the cell, the degree of sequence conservation, the genomic location —i.e., chromosome, chromid, plasmidome⁸— and different features such as codon ramps, and mRNA secondary structure among others³.

Measuring codon usage Bias

There are different kind of indices that quantify the codon bias of a gene. Some quantify the change in the frequency of use of each codon in relation to the frequency of random codon usage —e.g. Relative synonymous codon usage (RSCU, ENC)— while others compare to the codon usage bias observed on set of (predicted) highly expressed (PHE) proteins— e.g. Fop, CAI. Additionally, the translation adaptation index⁶ (tAI) determines the degree of adaptation of the codon usage to the amount of each tRNAs of the cell³.

And for a set of genes?

The obvious choice appears to be the average codon usage frequency of all the genes in the set. However, a method to calculate a frequency that represents in a more accurate way the codon usage frequency of the majority of the genes, denominated the modal frequency, was reported⁵. This method iteratively adjusts the codon usage frequency while comparing, by means of Chi-square statistics, to the codon usage frequency of all the genes in the set, until there is no statistically significant difference for most of the genes.

CUBES

CUBES is a set of linux bash and perl scripts which can be used to:

- 1** Calculate modal codon usage frequencies for sets of core-genomes (e.g. from a core-genome obtained from Edgar software)
 - Generate a representative DNA sequences for those modal frequencies (Since codon usage analysis will be done using codonW¹⁰, which requires coding sequences)
 - Calculate the correspondence analysis (COA) of RSCU using CodonW software.
 - Generate plots of the first 2 components of COA, both for genes and codons, including the modal sequences.
- 2** Calculate the variation of the adaptation index s-tAI and the GC3 content of the Core-genome Ci vs the evolutionary distance between the reference genome and the most distant genome added to generate the core-genome Ci. Requires a trna.txt file with the tRNA genes copy number for the reference species, and the evolutionary distance between the most distant genomes in the core-genome. The script outputs tables and plots showing the variation of these indexes in function of the evolutionary distance.
- 3** Generate a plot showing the change in the codon usage frequencies (CUF) for each codon and the corresponding adaptiveness w.
- 4** Generate a heatmap of the tRNA adaptiveness (w), the difference in CUF between the C1 (initial core-genome) and the most ancestral Core-genome (Cn), and between putatively highly expressed genes (PHE) and Cn.

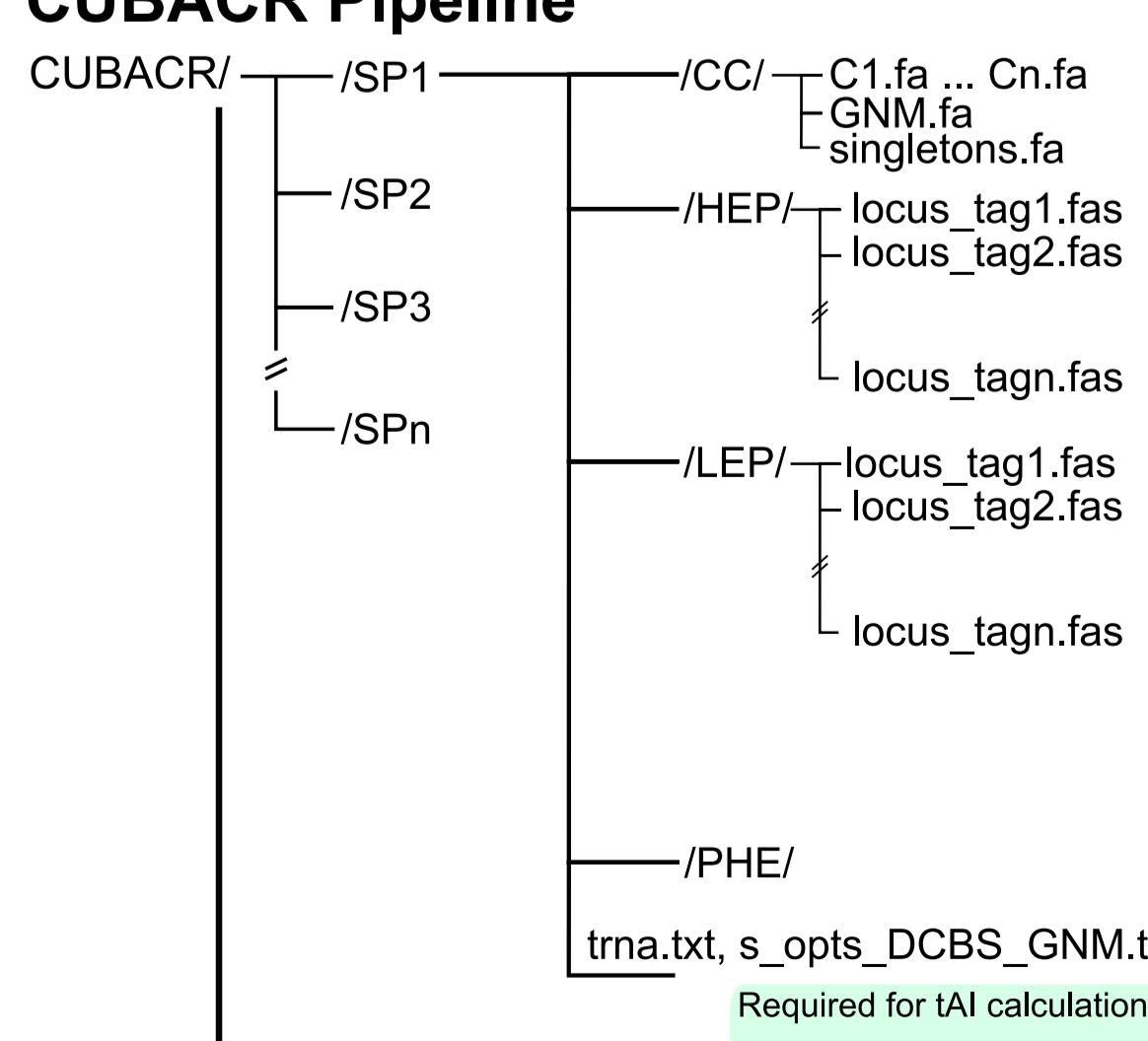
CUBACR

Scripts for the codon usage bias analysis of Conserved and Variable regions from core genome proteins. CUBACR scripts are programmed to process multifasta files containing orthologs for each gene of a desired core-genome set (or sets of proteins with a defined expression level), make an amino acid guided codon alignment for each gene (using TranslatorX software¹), and to output the conserved and variable regions of all genes of the set.

CUBACR can then be used to:

- 1** Make an amino acid based codon alignment and extract conserved and variable codons.
Calculate the modal codon usage frequencies for CR and VR.
Perform the correspondence analysis of RSCU for all genes and modal frequencies (Including CR and VR).
- 2** Calculate a neighbor joining tree using G. Olsen freqs_2_nj_tree_linux script⁹ (Type 2 distances between CUF. Requires Phylip⁷ software), and generate a heatmap of modal codon usage frequencies guided by the

CUBACR Pipeline



References

1. Abascal, F., Zardoya, R., & Reeder, M. J. Evolutionary divergence of codon usage patterns, shaped by different rates of transitions. *Nucleic Acids Res.* 38, 7–13 (2010).
2. Edgar, R. C. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nat Methods* 4, 357–360 (2007).
3. Camarero, G. & Schneider, A. Codon evolution – Mechanisms and Models. Oxford University Press (2012). doi:10.1093/oxfordhb/9780199562069.003.0001.
4. Contreras-Moreira, B. & Vieira, P. GET_HOMOLOGUES, a versatile software package for reliable and robust microbial pan genome analysis. *Appl Environ Microbiol* 79, 3613–3619 (2013).
5. Davis, J. J. & Olsen, G. J. Modal codon usage: Assessing the typical codon usage of a genome. *Mol Biol Evol* 27, 890–899 (2010).
6. dos Reis, M., Servi, R., & Wertheim, J. O. Solving the riddle of codon usage preferences: a test for translational selection. *Nucleic Acids Res.* 32, 5036–5044 (2004).
7. dos Reis, M., Servi, R., & Wertheim, J. O. Solving the riddle of codon usage preferences: a test for translational selection. *Nucleic Acids Res.* 32, 5036–5044 (2004).
8. López, J. L. et al. Codon Usage Heterogeneity in the Mammalian Prokaryote Genome: Selection-Based Coding Bias Associated with Gene Location, Expression Level, and Ancestry. *J Mol Evol* 31, 1–10 (2010).
9. Pedem, J. C. Correspondence Analysis of Codon Usage. Codonix (2005).
10. Team, R. C. R: A language and environment for statistical computing. (2016).

