scoup: Simulate Codon Sequences with Darwinian Selection Incorporated as an Ornstein-Uhlenbeck Process

Hassan Sadiq*
Department of Statistics and Actuarial Science, Stellenbosch University, South Africa.
Darren P. Martin
Institute of Infectious Diseases and Molecular Medicine, Division of Computational Biology, Department of Integrative Biomedical Sciences, University of Cape Town, South Africa.

28 February 2026

Summary

Genetic analyses of natural selection within and between populations have increasingly developed along separate paths. The two important genres of evolutionary biology (i.e. phylogenetics and population genetics) borne from the split can only benefit from research that seeks to bridge the gap. Simulation algorithms that combine fundamental concepts from both genres are important to achieve such unifying objective. We introduce `scoup`, a codon sequence simulator that is implemented in R and hosted on the Bioconductor platform. There is hardly any other simulator dedicated to genetic sequence generation for natural selection analyses on the platform. Concepts from the Halpern-Bruno mutation-selection model and the Ornstein-Uhlenbeck (OU) evolutionary algorithm were creatively fused such that the end-product is a novelty with respect to computational genetic simulation. Users are able to seamlessly adjust the model parameters to mimic complex evolutionary procedures that may have been otherwise infeasible. For example, it is possible to explicitly interrogate the concepts of static and changing fitness landscapes with regards to Darwinian natural selection in the context of codon sequences from multiple populations.

Statement of need

Statistical inference models that are used to analyse the degree of the impact of Darwinian natural selection on observed genetic data, command a healthy portion of the phylogenetic literature (Gupta and Vadde 2023). Validation of these largely codon-based models relies heavily on simulated data. Given the ever increasing diversity of natural selection inference models that exist (Yang 2007; Arenas 2015; Kosakovsky Pond et al. 2020), there is a need for more sophisticated simulators to match the expanding model complexities.

Bioconductor (Gentleman et al. 2004) is a leading bioinformatics platform distributing peer-reviewed R packages. A search of the entries on the platform, in Version 3.22 on 18 February 2026, with keywords including, `codon`, `mutation`, `selection`, `simulate`, and `simulation` returned a total of 70 packages (excluding `scoup`) out of the 2361 available. None of the retrieved entries was dedicated to codon data simulation for natural selection analyses. Thus, `scoup` is designed on the basis of the mutation-selection (MutSel) framework (Halpern and Bruno 1998) as an overdue contribution to the void.

Software and/or packages for simulating molecular protein sequences are a few in the scientific literature (Peng et al. 2015). Existing simulators tend to be more suitable for quantitative character evolution. These include, `ape` (Paradis and Schliep 2019), `ouch` (Cressler, Butler, and King 2015; Butler and King 2004) and `geiger` (Pennell et al. 2014). Other extensively used DNA sequence simulators including, `Seq-Gen` (Rambaut

*Corresponding author - hassan.t.sadiq@gmail.com

and Grass 1997), `INDELible` (Fletcher and Yang 2009), `PhyloSim` (Sipos et al. 2011) and `phangorn` (Schliep 2011) are parameterized in accordance with ω-based models (Goldman and Yang 1994; Muse and Gaut 1994). More recent sequence simulators, such as, `phastSim` (Maio et al. 2022) and `AliSim-HPC` (Ly-Trong, Barca, and Minh 2023) prioritized output capacity. Only few genetic simulators were built upon the more elaborate MutSel evolutionary concept. These include, `Pyvolve` (Spielman and Wilke 2015a) and `SGWE` (Arenas and Posada 2014). To the best of our knowledge, these existing MutSel friendly simulators are only able to generate data from static landscapes. With our proposed simulator, it is possible to generate codon sequences from landscapes that are static or those that are changing (also known as *seascapes*) (Mustonen and Lässig 2010).

Algorithm

`scoup` is further unique for at least three reasons. First, it incorporates Darwinian natural selection into the MutSel model in terms of variability of selection coefficients, an extension of an idea from Spielman and Wilke (2015b). Second, it directly utilises the concept of fitness landscapes. Third, fitness landscape updates can be executed in either a deterministic or a stochastic format. The stochastic updates are implemented in terms of the more biologically amenable, Ornstein-Uhlenbeck (OU) process (Bartoszek et al. 2017; Uhlenbeck and Ornstein 1930). A crude summary of how substitution events are executed in `scoup` is presented in Figure 1.

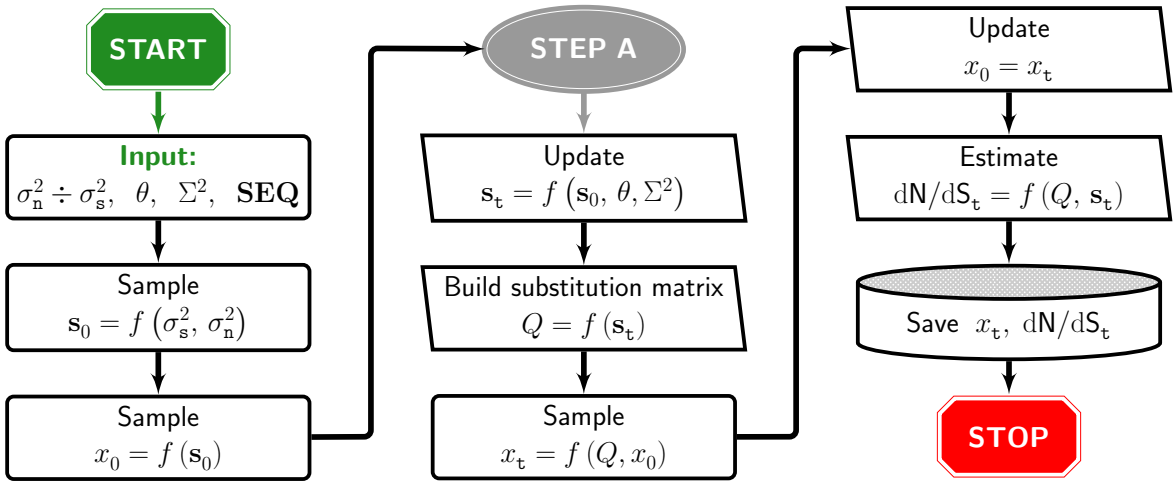

Figure 1: **Summarised `scoup` algorithm.** The flowchart shows the process for a single substitution event. After each substitution event, the process returns to *STEP A*, until the input tree length ($\tau \in$ **SEQ**) is exhausted. σ^2_n = variance of amino acid selection coefficients. σ^2_s = variance of synonymous codon selection coefficients. Σ^2 = OU asymptotic variance. θ = OU mean reversion rate. **SEQ** = sequence information. $x_\star$ = codon. $\mathbf{s}_\star$ = codon selection coefficient vector.

We highlight two important design choices from Figure 1. First, we assume that a static fitness landscape is obtained from a single set of parameters (ξ) needed to sample a 20-element numerical vector of amino acid selection coefficients (that is, s_0 in Figure 1). The coefficients are subsequently used as inputs of the corresponding MutSel model. The seascape setting is then defined as a function of multiple sets of parameters ($\xi_1, \xi_2, \ldots, \xi_k$, for $k \leq$ extant taxa size). Second, the coefficient update (s_t) step is done after every substitution event. In addition, the Ornstein-Uhlenbeck update process is discretised. In other words, the OU jump sizes are fixed and pre-specified as an input to the simulation functions.

Implementation

`scoup` is primarily designed using base functions in `R`. Some important complementary functions are imported

from the `Matrix` (Bates, Maechler, and Jagan 2024) and the `Biostrings` (Pagès et al. 2024) packages. We simulated some sequences with `scoup` to verify the accuracy of the outputs from the package. The output data each comprise eight sequences and 1000 codon sites. All the other necessary model parameters were kept the same for all simulated replicates. The data and all the associated files, including the simulation and analyses code, are available in the `paper/data` folder as part of the package. Only the variance of the selection coefficients of the synonymous codons ($\sigma_s^2 = \{0.00, 0.01\}$) and the variance of the amino acids ($\sigma_n^2 = \{0.001, 0.005, 0.009, 0.030, 0.070, 0.100, 0.500, 0.900\}$) were varied and five replicate sequences were generated for each (σ_s^2, σ_n^2) combination. The data sets were analysed with PAML (Yang 2007) to obtain maximum likelihood estimates of the ratio of the rates of non-synonymous to synonymous substitutions (ω) and these were compared to the analytical estimates ($\mathrm{d}N/\mathrm{d}S$) obtained from `scoup`. The results are summarised in Figure 2.

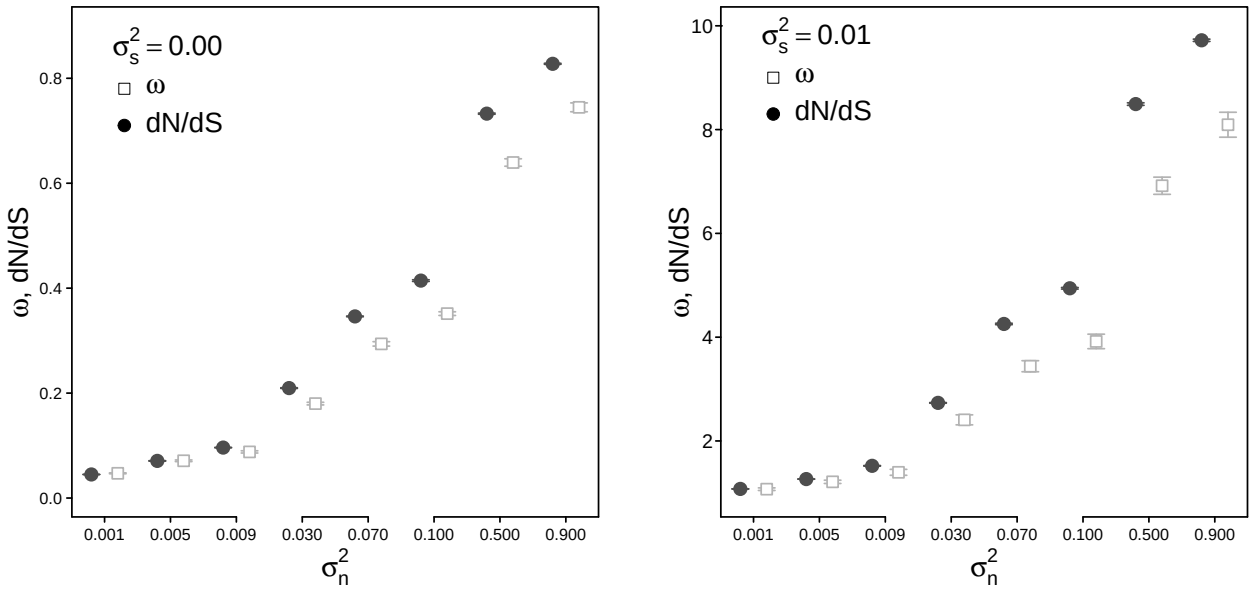

Figure 2: **Analyses of data generated with `scoup`.** The width of the arrows correspond to two times the associated standard errors. σ_n^2 = variance of amino acid selection coefficients. σ_s^2 = variance of synonymous codon selection coefficients. ω = maximum likelihood estimate of non-synonymous to synonymous substitution rates ratio obtained using PAML, $\mathrm{d}N/\mathrm{d}S$ = analytical equivalent of ω that is returned as part of the outputs from `scoup`.

Three features from Figure 2 are noteworthy. First, there is good correlation between the simulated (as measured by $\mathrm{d}N/\mathrm{d}S$) and the inferred (as measured by ω) magnitude of natural selection effect. The slight discrepancies as σ_n^2 increases are likely due to the limited sizes of the data sets (Spielman and Wilke 2015a). This imply that outputs from `scoup` are reliable. Second, as expected (Spielman and Wilke 2015b), in the absence of synonymous selection (that is, $\sigma_s^2 = 0$) the selection effect is predominantly negative (that is, $\mathrm{d}N/\mathrm{d}S$, $\omega < 1$) and the effect is largely positive when synonymous selection is present. This further asserts of the reliability of the outputs from `scoup`. Third, the magnitude of natural selection effect may be influenced by amino acid selection (or aptly, non-synonymous selection). This property is yet to be thoroughly investigated in the computational molecular evolution literature and there is hardly any other available computational resource that permits its exploration. This underlines the potential importance of `scoup`.

Conclusions

We present `scoup`, a R package for codon sequences simulation, where the evolutionary processes are mirrored more realistically than most existing simulators. Our framework creatively incorporates the Ornstein-Uhlenbeck process into the mutation-selection evolutionary model. This attribute could potentially unlock

exciting research avenues that will improve existing knowledge about the complex interactions of different, potentially interacting, molecular evolutionary processes.

Code availability

`scoup` is published for free public use under the GPL-2 license. It is available for download from the Bioconductor platform, along with detailed documentation and tutorial files. Some additional sample code are accessible in the `tests` and the `vignettes` folders of the package.

Acknowledgements

We thank Ben Murrell for suggesting modelling varying selection coefficients with an OU process. Computations were performed using the `HPC1` facility at Stellenbosch University, South Africa.

References

Arenas, Miguel. 2015. "Trends in Substitution Models of Molecular Evolution." *Frontiers in Genetics* 6 (319). https://doi.org/10.3389/fgene.2015.00319.

Arenas, Miguel, and David Posada. 2014. "Simulation of Genome-wide Evolution under Heterogeneous Substitution Models and Complex Multispecies Coalescent Histories." *Molecular Biology and Evolution* 31 (5): 1295–1301. https://doi.org/10.1093/molbev/msu078.

Bartoszek, Krzysztof, Sylvain Glémin, Ingemar Kaj, and Martin Lascoux. 2017. "Using the Ornstein–Uhlenbeck Process to Model the Evolution of Interacting Populations." *Journal of Theoretical Biology* 429: 35–45. https://doi.org/10.1016/j.jtbi.2017.06.011.

Bates, Douglas, Martin Maechler, and Mikael Jagan. 2024. *Matrix: Sparse and Dense Matrix Classes and Methods*. https://doi.org/10.32614/CRAN.package.Matrix.

Butler, Marguerite A, and Aaron A King. 2004. "Phylogenetic Comparative Analysis: A Modeling Approach for Adaptive Evolution." *The American Naturalist* 164 (6): 683–95. https://doi.org/10.1086/426002.

Cressler, Clayton E, Marguerite A Butler, and Aaron A King. 2015. "Detecting Adaptive Evolution in Phylogenetic Comparative Analysis Using the Ornstein-Uhlenbeck Model." *Systematic Biology* 64 (6): 953–68. https://doi.org/10.1093/sysbio/syv043.

Fletcher, William, and Ziheng Yang. 2009. "INDELible: A Flexible Simulator of Biological Sequence Evolution." *Molecular Biology and Evolution* 26 (8): 1879–88. https://doi.org/10.1093/molbev/msp098x.

Gentleman, Robert C, Vincent J Carey, Douglas M Bates, Ben Bolstad, Marcel Dettling, Sandrine Dudoit, Byron Ellis, et al. 2004. "Bioconductor: Open Software Development for Computational Biology and Bioinformatics." *Genome Biology* 5 (10): R80. https://doi.org/10.1186/gb-2004-5-10-r80.

Goldman, Nick, and Ziheng Yang. 1994. "A Codon-based Model of Nucleotide Substitution for Protein-coding DNA Sequences." *Molecular Biology and Evolution* 11 (5): 725–36. https://doi.org/10.1093/oxfordjournals.molbev.a040153.

Gupta, Manoj Kumar, and Ramakrishna Vadde. 2023. "Next-Generation Development and Application of Codon Model in Evolution." *Frontiers in Genetics* 14. https://doi.org/10.3389/fgene.2023.1091575.

Halpern, Aaron L, and William J Bruno. 1998. "Evolutionary Distances for Protein-Coding Sequences: Modelling Site-Specific Residue Frequencies." *Molecular Biology and Evolution* 15 (7): 910–17. https://doi.org/10.1093/oxfordjournals.molbev.a025995.

Kosakovsky Pond, Sergei L, Art F Y Poon, Ryan Velazquez, Steven Weaver, N Lance Hepler, Ben Murrell, Stephen D Shank, et al. 2020. "HyPhy 2.5 – A Customizable Platform for Evolutionary Hypothesis Testing Using Phylogenies." *Molecular Biology and Evolution* 37 (1): 295–99. https://doi.org/10.1093/molbev/msz197.

Ly-Trong, Nhan, Giuseppe M J Barca, and Bui Quang Minh. 2023. "AliSim-HPC: Parallel Sequence Simulator for Phylogenetics." *Bioinformatics* 39 (9): btad540. https://doi.org/10.1093/bioinformatics/btad540.

Maio, Nicola De, William Boulton, Lukas Weilguny, Conor R Walker, Yatish Turakhia, Russell Corbett-Detig, and Nick Goldman. 2022. "phastSim: Efficient Simulation of Sequence Evolution for Pandemic-Scale

Datasets." *PLOS Computational Biology* 18 (4): e1010056. https://doi.org/10.1371/journal.pcbi.1010056.

Muse, Spencer V, and Brandon S Gaut. 1994. "A Likelihood Approach for Comparing Synonymous and Nonsynonymous Nucleotide Substitution Rates, with Application to the Chloroplast Genome." *Molecular Biology and Evolution* 11 (5): 715–24. https://doi.org/10.1093/oxfordjournals.molbev.a040152.

Mustonen, Ville, and Michael Lässig. 2010. "Fitness Flux and Ubiquity of Adaptive Evolution." *Proceedings of the National Academy of Sciences* 107 (9): 4248–53. https://doi.org/10.1073/pnas.0907953107.

Pagès, Hervé, Patrick Aboyoun, Robert Gentleman, and Saikat DebRoy. 2024. *Biostrings: Efficient Manipulation of Biological Strings*. https://doi.org/10.18129/B9.bioc.Biostrings.

Paradis, Emmanuel, and Klaus Schliep. 2019. "ape 5.0: An Environment for Modern Phylogenetics and Evolutionary Analyses in R." *Bioinformatics* 35 (3): 526–28. https://doi.org/10.1093/bioinformatics/bty633.

Peng, Bo, Huann-Sheng Chen, Leah E Mechanic, Ben Racine, John Clarke, Elizabeth Gillanders, and Eric J Feuer. 2015. "Genetic Data Simulators and their Applications: An Overview." *Genetic Epidemiology* 39 (1): 2–10. https://doi.org/10.1002/gepi.21876.

Pennell, Matthew W, Jonathan M Eastman, Graham J Slater, Joseph W Brown, Josef C Uyeda, Richard G FitzJohn, Michael E Alfaro, and Luke J Harmon. 2014. "geiger v2.0: An Expanded Suite of Methods for Fitting Macroevolutionary Models to Phylogenetic Trees." *Bioinformatics* 30 (15): 2216--2218. https://doi.org/10.1093/bioinformatics/btu181.

Rambaut, Andrew, and Nicholas C Grass. 1997. "Seq-Gen: An Application for the Monte Carlo Simulation of DNA Sequence Evolution Along Phylogenetic Trees." *Bioinformatics* 13 (3): 235–38. https://doi.org/10.1093/bioinformatics/13.3.235.

Schliep, Klaus Peter. 2011. "phangorn: Phylogenetic Analysis in R." *Bioinformatics* 27 (4): 592--593. https://doi.org/10.1093/bioinformatics/btq706.

Sipos, Botond, Tim Massingham, Gregory E Jordan, and Nick Goldman. 2011. "PhyloSim – Monte Carlo Simulation of Sequence Evolution in the R Statistical Computing Environment." *BMC Bioinformatics* 12 (104). https://doi.org/10.1186/1471-2105-12-104.

Spielman, Stephanie J, and Claus O Wilke. 2015a. "Pyvolve: A Flexible Python Module for Simulating Sequences along Phylogenies." *PLoS ONE* 10 (9): e0139047. https://doi.org/10.1371/journal.pone.0139047.

———. 2015b. "The Relationship between dN/dS and Scaled Selection Coefficients." *Molecular Biology and Evolution* 32 (4): 1097–1108. https://doi.org/10.1093/molbev/msv003.

Uhlenbeck, G E, and L S Ornstein. 1930. "On the Theory of the Brownian Motion." *Physical Review* 36: 823–41. https://doi.org/10.1103/PhysRev.36.823.

Yang, Ziheng. 2007. "PAML 4: Phylogenetic Analysis by Maximum Likelihood." *Molecular Biology and Evolution* 24 (8): 1586–91. https://doi.org/10.1093/molbev/msm088.