Application of mechanistic models to separate the effects of mutation, selection, and drift on protein sequence evolution

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Cedric Lars Florian Landerer

December 2018

To my mother

Acknowledgments

I am grateful for the many people at the University of Tennessee and in Knoxville that that made my time here such a pleasure. First and foremost I want to thank my Adviser, Dr. Michael Gilchrist for his long lasting patience, his availability and his teachings; always sharpening my focus and providing a new angle to a problem. Great thanks also goes to my committee Dr. Benjamin Fitzpatrick, Dr. Brian O'Meara, and Dr. Russel Zaretzki as they were always available for questions and discussions and for their great guidance. In particular Brian O'Meara who always had an open door and tolerated my frequent visits. None of the work presented in this dissertation would have been possible without their great guidance. For many great discussions and never a dull momment in the office I also have to thank my labmate Alex Cope. I also have to thank the faculty and students in Ecology and Evolutionary Biology, allowing me to broaden my knowledge and insights with always stimulating discussions and for moral support. Specially John Reese, Cassie Dresser, Liam Muller, Athmanathan Senthilnathan, Harmony Yomai, and Jim Fordyce. Thanks also goes to my former roommate Cassie Watters without whom my stay in Knoxville would have been a lot less exciting.

Nothing in Biology Makes Sense Except in the Light of Evolution.

-Theodosius Dobzhansky

Nothing in evolutionary biology makes sense except in the light of population genetics

-Michael Lynch

Abstract

Mathematical and statistical models are useful for describing and understanding observations in genetics and genomics. These models have to constantly be updated to reflect current biological understanding. As opposed to descriptive and phenomenological models, mechanistic models allow for the extraction of more biologically relevant information based on underlying principles. Mutation, selection, and genetic drift are the three forces guiding evolution. Mechanistic models rooted in population genetics principles allow us to determine how these forces shape observed data. I demonstrate the usage of mechanistic models to relate protein coding sequences to their fitness landscapes and the evolutionary forces shaping them. Using the yeast *L. kluyveri*, I show the increased cost of protein synthesis due to a large scale introgression with mismatched codon usage. Furthermore, I analyze site-specific selection on amino acids in the beta-lactamase protein TEM, which confers antibiotic resistance in *E. coli* and related species.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Protein synthesis is the most costly metabolic process a cell performs (Reeds *et al.*, 1985; Waterlow and Millward, 1989; Buttgereit and Brand, 1995; Warner, 1999; Akashi and Gojobori, 2002; Lindqvist *et al.*, 2018) causing selection to maximize the benefit of protein synthesis and performing it as efficiently as possible. Studying the ratio of cost to benefit of protein synthesis is, therefore, important to understand the evolution of protein coding sequences (Gilchrist *et al.*, 2009; Shah and Gilchrist, 2011a; Gilchrist *et al.*, 2015; Beaulieu *et al.*, 2019). However, the strength of selection varies greatly between genes, from low expression genes with codon usage dominated by mutation bias between nucleotides over highly expressed genes reflecting the dominance of selection for efficient translation of the mRNA, to selection on the amino acid composition required for the function of the protein.

We can formalize the cost and benefit of a protein coding sequence and formulate mathematical models. Mathematical and statistical models have long been used to describe or summarize observations in genetics and genomics. Often without addressing the underlying biological mechanisms - mutation, selection, and genetic drift - shaping DNA sequences, but as phenomelogical descriptions. As researchers learn more about the underlying processes and more genetic and genomic data is available, the mathematical models that allow for the extraction of information from this data have to keep up. For example, after the unraveling of the degenerate genetic code by Matthaei and Nierenberg

(1961); Nierenberg and Matthaei (1961); Maxwell (1962); Leder and Nierenberg (1964), and many others, researchers noticed that synonymous codons are not found in uniform proportions (Fitch, 1976; Grantham *et al.*, 1980; Ikemura, 1981; Grantham *et al.*, 1981; Sharp *et al.*, 1988). Models of codon usage, however, were long purely descriptive and heuristic (Ikemura, 1981; Bennetzen and Hall, 1982; Sharp and Li, 1987; Wright, 1990). Similarly, phylogenetic models have long been phenomelogical (Jukes and Cantor, 1969; Dayhoff *et al.*, 1978; Kimura, 1980; Felsenstein, 1981; Altschul, 1991), describing the rate of change between states without regards for the forces guiding evolution, mutation, selection, and genetic drift. Zuckerkandl and Pauling (1962) proposed that the evolution of proteins is constant over time and between lineages before the genetic code was fully deciphered and at a time were protein synthesis was barely understood based on their observation that similarity on hemoglobin is correlated with divergence time. This work is therefore focused on the application of mechanistic models rooted in first principles and their application to protein coding sequences

Mechanistic models are used throughout biology (Goldman and Yang, 1994; Laureau, 1998; Davis and Pelsor, 2001; Doron-Faigenboim and Pupko, 2007; McGill *et al.*, 2007). By modeling the process underlying the observed data mechanistic models provide insights into the processes and estimates of parameters shaping the data (Liberles *et al.*, 2013). A wide variety of information is stored in protein and protein coding sequences, e.g. structure (Anfinsen, 1973), mutation bias (Shah and Gilchrist, 2011a; Gilchrist *et al.*, 2015), protein synthesis rate (Gilchrist, 2007; Gilchrist *et al.*, 2015). Mechanistic models can be used to extract these informations and to study the relative strength of mutation, selection, and genetic drift leading to the observed sequences.

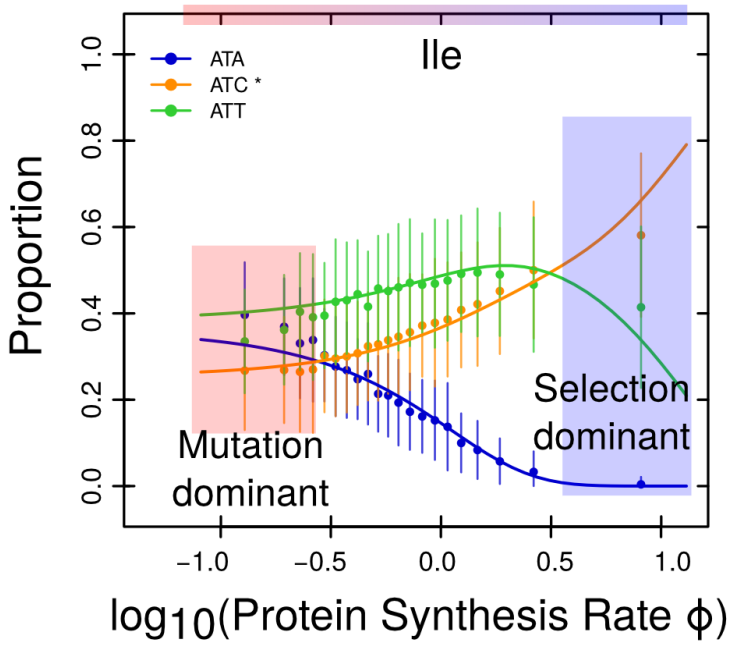


Figure 1.1: ROC SEMPPR model behavior for Isoleucine. The proportion of each codon observed changes with protein synthesis rate. Mutation is dominant when protein synthesis rate is low, mutationally favored codons are observed with the highest frequency. With the increase of protein synthesis rate, the influence of selection increases until the system is dominated by selection. The selectively favored codon is observed with the highest frequency.

1.1 Cost: Decomposing Codon Usage

Mutation bias on codon usage is a reflection of the cellular environment while selection on codon usage allows us to make inferences about the cellular and external environment a genome has evolved in. The relative strength of mutation and selection on individual genes varies, allowing us to separate mutation bias and selection, specifically selection against translation overhead cost (Gilchrist, 2007; Shah and Gilchrist, 2011a; Gilchrist *et al.*, 2015). Genes with low protein synthesis rates are thought to be under weak selection for codon usage and their codon usage is therefore dominated by mutation bias. In contrast, genes with high protein synthesis rates are thought to be under strong selection and their codon usage is therefore dominated by selection. However, mutation bias and selection can differ within the genome.

For example, strand specific mutation bias (Lafay *et al.*, 1999; Romero *et al.*, 2000), differences in the tRNA pool throughout life stages (Sagi *et al.*, 2016), or introgressions and horizontal gene transfer (Mdigue *et al.*, 1991; Lawrence and Ochman, 1997) can produce multiple genomic environments. Chapter 2 extends the mechanistic model ROC SEMPPR Gilchrist *et al.* (2015) to allow for a mixture distribution of mutation and selection parameters Landerer *et al.* (2018) and provides researchers with a software tool to address intra genomic variation in codon usage. However, there is a significant difference to classical mixture approaches. In addition to gene population specific parameters, ROC SEMPPR also estimates a gene specific parameter (protein synthesis rate). Therefore, the protein synthesis rate for each gene has to be estimated assuming that the a gene is in each gene population. This can provide additional insight into the adaptiveness of a gene to alternative genomic environments. Figure 1.1 illustrates how the proportions of synonymous codons change with increasing protein synthesis rate. When the protein synthesis rate is low, mutation bias between codons dominates the proportions of synonymous codons while increasing protein synthesis increases the strength of selection (see Gilchrist *et al.* (2015) for details).

In chapter 3, I apply AnaCoDa to analyze the synonymous codon usage of the yeast *L. kluyveri* which experienced a large scale introgression replacing the whole left arm of chromosome C (Friedrich *et al.*, 2015). I studied the differences in the parameters describing codon usage between the endogenous *L. kluyveri* genes and the introgressed exogenous genes. Recognizing the differences in codon usage between the endogenous and exogenous genes allowed me to improve prediction of protein synthesis rate, and separate the effects of mutation bias and selection in the endogenous *L. kluyveri* genes and the introgressed exogenous genes. This information was used to determine a potential donor lineage in *E. gossypii*, estimate the time since introgression, and estimate the genetic load of the introgression.

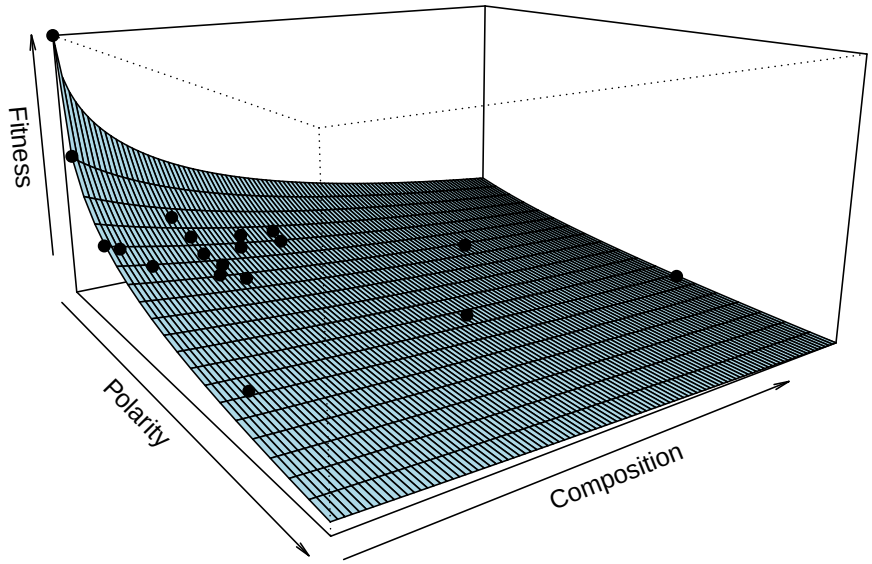


Figure 1.2: Decline in fitness with distance in physicochemical space from the optimal amino acid. Fitness decline of amino acids (black dots) relative to optimal amino acid (Alanine). Weighting of physicochemical properties according to Grantham (1974). The full fitness surface can be described but only 20 discrete amino acid states are available for selection to act on.

1.2 Benefit: Selection on Amino acids

Genes are evolving with natural selection favoring proteins that encode their function optimally, with mutations and genetic drift reducing functionality. Amino acid preference and the relative strength of mutation, selection, and genetic drift usually varies between sites along the protein sequence. The number of parameters required to describe protein fitness increases exponentially with the length of the protein if interactions between sites are accounted for. Attempts to incorporate selection into phylogenetic approaches are, therefore, limited to site specific selection. The goal of chapter 4 is to estimate the strength of site specific selection on amino acids from protein coding sequences in a phylogenetic framework.

Ignoring interactions between sites allows to describe the site specific fitness landscape of a protein. Some approaches rely on the description of the full fitness landscape and therefore require $19 \times L$, where L is the length of the peptide in amino acids, parameters (Lartillot and Philippe, 2004; Le *et al.*, 2008; Wang *et al.*, 2008; Holder *et al.*,

2008; Wu *et al.*, 2013; Tamuri *et al.*, 2014). As this is still a large number of parameters the incorporation of experimentally determined site specific selection on amino acids is an attractive alternative (Bloom, 2014; Thyagarajan and Bloom, 2014; Bloom, 2017). Alternatively, assumptions about the nature of selection can reduce the number of parameters required. For example, frequency dependent selection (Goldman and Yang, 1994; Muse and Gaut, 1994; Thorne *et al.*, 1996) or stabilizing selection (Beaulieu *et al.*, 2019) allow for a reduction in fitness of amino acids with distance in physicochemical space.

SelAC (Beaulieu *et al.*, 2019) is a model of stabilizing selection that assesses the fitness of each amino acid relative to the fitness peak (Figure 1.2). The ftness of an amino acid is assumed to decline exponentially with distance to the optimal amino acid in physicochemical space. In chapter 4 I apply *SelAC* to the β-lactamase TEM and estimate site specific selection on amino acids and compare the inferred fitness landscape to empirical estimates from deep mutation scanning experiments (Stiffler *et al.*, 2016). I find that experimentally informed amino acid preferences improve model fit but do not accurately reflect the evolution of TEM in the wild. Furthermore, I show that the information on site specific selection on amino acids can be extracted from protein coding sequences by models rooted in first principles like *SelAC*.

Chapter 2

AnaCoDa: Analyzing Codon Data with Bayesian mixture models

This chapter is a lightly revised version of a paper by the same name published in Bioinformatics and co-authored with Alexander Cope, Russell Zaretzki, and Michael A. Gilchrist.

C. Landerer, A. Cope, R. Zaretzki, M.A. Gilchrist, AnaCoDa: analyzing codon data with Bayesian mixture models, Bioinformatics, 34, 2018, 2496-2498

2.1 Abstract

AnaCoDa is an R package for estimating biologically relevant parameters of mixture models, such as selection against translation inefficiency, nonsense error rate, and ribosome pausing time, from genomic and high throughput datasets. **AnaCoDa** provides an adaptive Bayesian MCMC algorithm, fully implemented in C++ for high performance with an ergonomic R interface to improve usability. **AnaCoDa** employs a generic object-oriented design to allow users to extend the framework and implement their own models. Current models implemented in **AnaCoDa** can accurately estimate biologically relevant parameters given either protein coding sequences or ribosome foot-printing data. Optionally, **AnaCoDa** can utilize additional data sources, such as gene expression measurements, to aid model fitting and parameter estimation. By utilizing a hierarchical object structure, some parameters can vary between sets of genes while others can be shared. Genes may be assigned to clusters or membership may be estimated by **AnaCoDa**. This flexibility allows users to estimate the same model parameter under different biological conditions and categorize genes into different sets based on shared model properties embedded within the data. **AnaCoDa** also allows users to generate simulated data which can be used to aid model development and model analysis as well as evaluate model adequacy. Finally, **AnaCoDa** contains a set of

visualization routines and the ability to revisit or re-initiate previous model fitting, providing researchers with a well rounded easy to use framework to analyze genome scale data.

Availability:

AnaCoDa is freely available under the Mozilla Public License 2.0 on CRAN (https://cran.r-project.org/web/packages/AnaCoDa/).

2.2 Introduction

AnaCoDa is an open-source software implemented in R (R Core Team, 2015) that allows researchers to analyze genome-scale data like coding sequences and ribosome footprinting data using evolutionary or analytical models in a Bayesian framework. **AnaCoDa** was developed to analyze selection on synonymous codon usage in the form of ribosome overhead cost (Gilchrist *et al.*, 2015; Wallace *et al.*, 2013; Shah and Gilchrist, 2011b). However, other codon metrics like the codon adaptation index (Sharp and Li, 1987) or the effective number of codons (Wright, 1990) are also provided as reference. In addition, three currently unpublished models to analyze coding sequences for evidence of selection against nonsense errors and estimate ribosome pausing times from ribosome footprinting data are included. **AnaCoDa** implements an adaptive Gibbs sampler within a Metropolis-Hastings Monte Carlo Markov Chain (MCMC). This allows for the incorporation of prior knowledge such as observed gene expression levels and easy sampling from the posterior distribution to estimate parameter values and quantify degree of uncertainty. **AnaCoDa** provides a mixture distribution option to all implemented models, combining genes into sets by estimating the posterior probabilities of set membership based on gene-set specific parameters shared by all genes assigned to a given set. **AnaCoDa** provides a generic, mixture distribution option to all implemented models, allowing for the estimation of condition specific parameters or the automatic categorization of data into different sets based on differences in their posterior probabilities of set membership. In addition to the four models provided, **AnaCoDa** provides a modular infrastructure such that additional genome scale or even phylogenetic models can be integrated.

The **AnaCoDa** framework works with **AnaCoDa** requires gene specific data such as codon frequencies obtained from coding sequences or position specific footprint counts. Conceptually, **AnaCoDa** allows for three different types of parameters. The first type are gene specific parameters such as protein synthesis rate or relative functionality. The second type are gene-set specific parameters, such as mutation bias terms or translation error rates.

These parameters are shared across genes within a set and can be exclusive to a single set or shared with other sets. While the number of gene sets must be pre-defined by the user, set assignment of genes can be pre-defined or estimated as part of the model fitting. Estimation of the set assignment provides the probability of a gene being assigned to a set allowing the user to asses the uncertainty in each assignment. The third type are hyperparameters allowing for the construction and analysis of hierarchical model. Hyperparameters control the prior distribution for gene and gene-set specific parameters such as mutation bias or protein synthesis rate.

2.3 Features

AnaCoDa provides an interface written in R, a freely available programming language noted for its ease of use for even inexperienced programmers. As a result, **AnaCoDa** is accessible to researchers with minimal computational experience.

The interface of **AnaCoDa** is designed for quick and efficient data analysis. Generally, the only input needed for fitting a model to the data are protein-coding codon sequences in the form of a FASTA file or a flat-file containing codon counts obtained from ribosome foot-printing experiments. **AnaCoDa** also provides visualization functionality, including plotting functions to compare parameter estimates for different mixture distributions and display codon usage patterns. In addition, diagnostic functions such as those for calculating and visualizing the degree of autocorrelation in the parameter traces are provided.

Robust and efficient model fitting

AnaCoDa has built-in features designed to improve the robustness and performance of the implemented MCMC approach. For example, the implemented MCMC automatically adapts the proposal width for sampled parameters such that an user defined acceptance range is met, improving sampling efficiency of the MCMC and computational performance. Even though

AnaCoDa is written in C++, analysis of large datasets and/or complex models can be very computationally intensive. To protect users from computer failures or aid in the collection of additional MCMC samples, **AnaCoDa** can periodically produce output checkpoint files, which can be used to restart an MCMC chain from a previous time point. In addition, **AnaCoDa** automatically thins all parameter traces - meaning only every k^{th} sample is kept - increasing the effective number of samples and reducing its memory footprint.

Although **AnaCoDa** is provided as an R package, the main computational work is implemented in C++. Because R does not provide native C++ support, Rcpp was employed to expose whole C++ classes as modules to R (Edelbuettel and Francois, 2011). Using Rcpp eliminates time consuming data transfers between the R environment and the C++ core during model fitting, resulting in improved computational performance and allows for a fully object-oriented code design (Booch, 1993). As expected, the runtime of **AnaCoDa** scales linearly with genome size and number of iterations, and scales polynomially with the number of mixture distributions in the data set. The polynomial increase in runtime with the number of mixture distributions is due to the necessity to condition the gene assignment on the estimation of gene specific parameters, such as, protein synthesis rate.

Data Simulation

In addition to fitting the models to datasets, **AnaCoDa** can be used to generate simulated data sets as well. On their own, simulated datasets are useful for model development and analysis. Simulating data under different conditions allows the user to explore model behavior and explore theoretical scenarios. Different conditions can include the addition or elimination of parameters, or simply allowing a set of parameter values to vary. Fitting models to simulated data can provide insight into potential pitfalls or shortcomings when fitting observational data and can serve as the basis for evaluating model adequacy of a model fit to observational data (Mi *et al.*, 2015). Significant differences between simulated

and observational data suggests the current set of parameters or the model as a whole fail to include or adequately represent biological mechanisms underlying the observed data.

Available models

AnaCoDa currently provides codon models for analyzing genome scale data. The ROC model implements and extends the codon usage bias (CUB) models developed by Gilchrist *et al.* (2015); Wallace *et al.* (2013); Shah and Gilchrist (2011b), which can reliably estimate the strength of selection on ribosome overhead cost, mutation bias and allows for the inference of protein synthesis rates. This model allows for the separation of effects of mutation and selection based on gene ordering by protein synthesis rate, and the addition of a mixture distribution allows for gene clustering based on mutation bias and selection for translation efficiency. In addition to identifying the most efficient codons, ROC provides estimates of mutation bias allowing the approximation of mutation ratios between codons (Gilchrist *et al.*, 2015; Wallace *et al.*, 2013).

The ability to estimate protein synthesis rates in the absence of empirical data is useful for investigating CUB of non-model organisms for which such data is lacking and enables the usage of protein synthesis rate in comparative frameworks or other analyses requiring protein synthesis rate information (Dunn *et al.*, 2018). Use of the mixture model allows for the investigation of CUB heterogeneity at the genome or gene level. Following the same framework, additional models included in **AnaCoDa** provide estimates of codon-specific nonsense errors rates (FONSE) and ribosome pausing times (PA and PANSE).

Parameters estimated with the evolutionary models ROC and FONSE represent evolutionary averages and do not depend on experimental conditions. In contrast, PA and PANSE estimate the distribution of biologically relevant parameters like ribosome pausing times along a gene from experimental data such as ribosome footprinting data. The distribution can be dependent (PANSE) or independent (PA) of evidence for nonsense errors in the data.

2.4 Appendix: Supplementary Material

AnaCoDa allows for the estimation of biologically relevant parameters like mutation bias or ribosome pausing time, depending on the model employed. Bayesian estimation of parameters is performed using an adaptive Metropolis-Hasting within Gibbs sampling approach. Models implemented in AnaCoDa are currently able to handle gene coding sequences and ribosome footprinting data.

2.4.1 The AnaCoDa framework

The AnaCoDa framework works with gene specific data such as codon frequencies or position specific footprint counts. Conceptually, AnaCoDa uses three different types of parameters.

- The first type of parameters are **gene specific parameters** such as gene expression level or functionality. Gene-specific parameters are estimated separately for each gene and can vary between potential gene categories or sets.

- The second type of parameters are **gene-set specific parameters**, such as mutation bias terms or translation error rates. These parameters are shared across genes within a set and can be exclusive to a single set or shared with other sets. While the number of gene sets must be pre-defined by the user, set assignment of genes can be pre-defined or estimated as part of the model fitting. Estimation of the set assignment provides the probability of a gene being assigned to a set allowing the user to asses the uncertainty in each assignment.

- The third type of parameters are **hyperparameters**, such as parameters controlling the prior distribution for mutation bias or error rate. Hyperparameters can be set specific or shared across multiple sets and allow for the construction and analysis of hierarchical models, by controlling prior distributions for gene or gene-set specific parameters.

Analyzing protein coding gene sequences

AnaCoDa always requires the following four objects:

- **Genome** contains the codon data read from a fasta file as well as empirical protein synthesis rate in the form of a comma separated (.csv) ID/Value pairs.
- **Parameter** represents the parameter set (including parameter traces) for a given genome. The parameter object also hold the mapping of parameters to specified sets.
- **Model** allows you to specify which model should be applied to the genome and the parameter object.
- **MCMC** specifies how many samples from the posterior distribution of the specified model should be stored to obtain parameter estimates.

2.4.2 AnaCoDa setup

Application of codon model to single genome

In this example we are assuming a genome with only one set of gene-set specific parameters, hence **num.mixtures** = 1. We assign all genes the same gene-set, and provide an initial value for the hyperparameter sphi (s_ϕ). s_ϕ controls the lognormal prior distribution on the gene specific parameters like the protein synthesis rate ϕ. To ensure identifiability the expected value of the prior distribution is assumed to be 1.

$$E[\phi] = \exp\left(m_\phi + \frac{s_\phi^2}{2}\right) = 1 \tag{2.1}$$

Therefore the mean m_ϕ is set to be $-\frac{s_\phi^2}{2}$. For more details see Gilchrist *et al.* (2015).

After choosing the model and specifying the necessary arguments for the MCMC routine, the MCMC is run

```
genome <- initializeGenomeObject(file = "genome.fasta")
```

```
parameter <- initializeParameterObject(genome = genome, sphi = 1,
                          num.mixtures = 1,
                          gene.assignment = rep(1, length(genome)))
model <- initializeModelObject(parameter = parameter, model = "ROC")
mcmc <- initializeMCMCObject(samples = 5000, thinning = 10,
                          adaptive.width=50)
runMCMC(mcmc = mcmc, genome = genome, model = model)
```

runMCMC does not return a value, the results of the MCMC are stored automatically in the mcmc and parameter objects created earlier.

> **Please note that AnaCoDa utilizes C++ object orientation and therefore employs pointer structures. This means that no return value is necessary for such objects as they are modified within the the runMCMC routine. You will find that after a completed run, the parameter object will contain all necessary information without being directly passed into the MCMC routine. This might be confusing at first as it is not default R behavior.**

Application of codon model to a mixture of genomes

This case applies if we assume that parts of the genome differ in their gene-set specific parameters. This could be due to introgression events or strand specific mutation difference, horizontal gene transfers or other reasons. We make the assumption that all sets of genes are independent of one another. For two sets of gene-set specific parameter with a random gene assignment we can use:

```
parameter <- initializeParameterObject(genome = genome,
                          sphi = c(0.5, 2), num.mixtures = 2,
                          gene.assignment = sample.int(2,
                                  length(genome), replace = T))
gene.assignment = sample.int(2, length(genome), replace = T))
```

To accommodate for this mixing we only have to adjust **sphi**, which is now a vector of length 2, **num.mixtures**, and **gene.assignment**, which is chosen at random here.

Empirical protein synthesis rate values

To use empirical values as prior information one can simply specify an observed.expression.file when initializing the genome object.

```
genome <- initializeGenomeObject(file = "genome.fasta",
                observed.expression.file = "synthesis_values.csv")
```

These observed expression or synthesis values (Φ) are independent of the number of gene-sets. The error in the observed Φ values is estimated and described by sepsilon (s_ϵ). The csv file can contain multiple observation sets separated by comma. For each set of observations an initial s_ϵ has to be specified.

```
# One case of observed data
sepsilon <- 0.1
# Two cases of observed data
sepsilon <- c(0.1, 0.5)
# ...
# Five cases of observed data
sepsilon <- c(0.1, 0.5, 1, 0.8, 3)
parameter <- initializeParameterObject(genome = genome, sphi = 1,
                          num.mixtures = 1,
                          gene.assignment = rep(1, length(genome)),
                          init.sepsilon = sepsilon)
```

In addition one can choose to keep the noise in the observations (s_ϵ) constant by using the **fix.observation.noise** flag in the model object.

```
model <- initializeModelObject(parameter = parameter, model = "ROC",
fix.observation.noise = TRUE)
```

Fixing parameter types

It can sometime be advantages to fix certain parameters, like the gene specific parameters. For example in cases where only few sequences are available but gene expression measurements are at hand we can fix the gene specific parameters to increase confidence in our estimates of gene-set specific parameters.

We again initialize the **genome**, **parameter**, and **model** objects.

```
genome <- initializeGenomeObject(file = "genome.fasta")
parameter <- initializeParameterObject(genome = genome, sphi = 1,
                          num.mixtures = 1,
                          gene.assignment = rep(1, length(genome)))
model <- initializeModelObject(parameter = parameter, model = "ROC")
```

To fix gene specific parameters we will set the **est.expression** flag to **FALSE**. This will estimate only gene-set specific parameters, hyperparameters, and the assignments of genes to various sets.

```
mcmc <- initializeMCMCObject(samples, thinning=1,
                          adaptive.width=100, est.expression=FALSE,
                          est.csp=TRUE, est.hyper=TRUE, est.mix=TRUE)
```

If we would like to fix gene-set specific parameters we instead disable the **est.csp** flag.

```
mcmc <- initializeMCMCObject(samples, thinning=1,
                          adaptive.width=100, est.expression=TRUE,
                          est.csp=FALSE, est.hyper=TRUE, est.mix=TRUE)
```

The same applies to the hyper parameters (**est.hyper**),

```
mcmc <- initializeMCMCObject(samples, thinning=1,
                          adaptive.width=100, est.expression=TRUE,
                          est.csp=TRUE, est.hyper=FALSE, est.mix=TRUE)
```

and gene set assignment (**est.mix**).

```
mcmc <- initializeMCMCObject(samples, thinning=1,
                             adaptive.width=100, est.expression=TRUE,
                             est.csp=TRUE, est.hyper=TRUE, est.mix=FALSE)
```

We can use these flags to fix parameters in any combination.

Combining various gene-set specific parameters to a gene-set description.

We distinguish between three simple cases of gene-set descriptions, and the ability to customize the parameter mapping. The specification is done when initializing the parameter object with the **mixture.definition** argument.

We encounter the simplest case when we assume that all gene sets are independent.

```
parameter <- initializeParameterObject(genome = genome,
                             sphi = c(0.5, 2), num.mixtures = 2,
                             gene.assignment = sample.int(2,
                                     length(genome), replace = T),
                             mixture.definition = "allUnique")
```

The **allUnique** keyword allows each type of gene-set specific parameter to be estimated independent of parameters describing other gene sets.

In case we want to share mutation parameter between gene sets we can use the keyword **mutationShared**

```
parameter <- initializeParameterObject(genome = genome,
                             sphi = c(0.5, 2), num.mixtures = 2,
                             gene.assignment = sample.int(2,
                                     length(genome), replace = T),
                             mixture.definition = "mutationShared")
```

This will force all gene sets to share the same mutation parameters.

The same can be done with parameters describing selection, using the keyword **selectionShared**

```
parameter <- initializeParameterObject(genome = genome,
                          sphi = c(0.5, 2), num.mixtures = 2,
                          gene.assignment = sample.int(2,
                                  length(genome), replace = T),
                          mixture.definition = "selectionShared")
```

For more intricate compositions of gene sets, one can specify a custom $n \times 2$ matrix, where n is the number of gene sets, to describe how gene-set specific parameters should be shared. Instead of using the **mixture.definition** argument one uses the **mixture.definition.matrix** argument.

The matrix representation of **mutationShared** can be obtained by

```
# [,1] [,2]
#[1,] 1 1
#[2,] 1 2
#[3,] 1 3
defMatrix <- matrix(c(1,1,1,1,2,3), ncol=2)
parameter <- initializeParameterObject(genome = genome,
                          sphi = c(0.5, 2, 1), num.mixtures = 3,
                          gene.assignment = sample.int(3,
                                  length(genome), replace = T),
                          mixture.definition.matrix = defMatrix)
```

Columns represent mutation and selection, while each row represents a gene set. In this case we have three gene sets, each sharing the same mutation category and three different selection categories. In the same way one can produce the matrix for three independent gene sets equivalent to the **allUnique** keyword.

```
# [,1] [,2]
```

```
#[1,] 1 1
#[2,] 2 2
#[3,] 3 3
defMatrix <- matrix(c(1,2,3,1,2,3), ncol=2)
```

We can also use this matrix to produce more complex gene set compositions.

```
# [,1] [,2]
#[1,] 1 1
#[2,] 2 1
#[3,] 1 2
defMatrix <- matrix(c(1,2,1,1,1,2), ncol=2)
```

In this case gene set one and three share their mutation parameters, while gene set one and two share their selection parameters.

Checkpointing

AnaCoDa does provide checkpointing functionality in case runtime has to be restricted. To enable checkpointing, one can use the function **setRestartSettings**.

```
# writing a restart file every 1000 samples
setRestartSettings(mcmc, "restart_file", 1000, write.multiple=TRUE)
# writing a restart file every 1000 samples
# but overwriting it every time
setRestartSettings(mcmc, "restart_file", 1000, write.multiple=FALSE)
```

To re-initialize a parameter object from a restart file one can simply pass the restart file to the initialization function:

```
initializeParameterObject(init.with.restart.file="restart_file.rst")
```

Load and save parameter objects

AnaCoDa is based on C++ objects using the Rcpp (Edelbuettel and Francois, 2011). This comes with the problem that C++ objects are by default not serializable and can therefore not be saved/loaded with the default R save/load functions.

AnaCoDa however, does provide functions to load and save parameter and mcmc objects. These are the only two objects that store information during a run.

```
#save objects after a run
runMCMC(mcmc = mcmc, genome = genome, model = model)
writeParameterObject(parameter = parameter, file = "parameter.Rda")
writeMCMCObject(mcmc = mcmc, file = "mcmc_out.Rda")
```

As **genome**, and **model** objects are purely storage containers, no save/load function is provided at this point, but will be added in the future.

```
#load objects
parameter <- loadParameterObject(file = "parameter.Rda")
mcmc <- loadMCMCObject(file = "mcmc_out.Rda")
```

2.4.3 File formats

Protein coding sequence

Protein coding sequences are provided by fasta file with the default format. One line containing the sequence id starting with > followed by the id and one or more lines containing the sequence. The sequences are expected to have a length that is a multiple of three. If a codon can not be recognized (e.g AGN) it is ignored.

```
>YAL001C
TTGGTTCTGACTCATTAGCCAGACGAACTGGTTCAA
CATGTTTCTGACATTCATTCTAACATTGGCATTCAT
```

```
ACTCTGAACCAACTGTAAGACCATTCTGGCATTTAG
>YAL002W
TTGGAACAAAACGGCCTGGACCACGACTCACGCTCT
TCACATGACACTACTCATAACGACACTCAAATTACT
TTCCTGGAATTCCGCTCTTAGACTCAACTGTCAGAA
```

Empirical gene expression

Empirical expression or gene specific parameters are provided in a csv file format. The first line is expected to be a header describing each column. The first column is expected be the gene id, and every additional column is expected to be represent a measurement. Each row corresponds to one gene and contains all measurements for that gene, including missing values.

```
>YAL001C
ORF,DATA_1,DATA_2,...DATA_N
YAL001C,0.254,0.489,...,0.156
YAL002W,1.856,1.357,...,2.014
YAL003W,10.45,NA,...,9.564
YAL005C,0.556,0.957,...,0.758
```

Ribosome foot-printing counts

Ribosome foot-printing (RFP) counts are provided in a csv file format. The first line is expected to be a header describing each column. The columns are expected in the following order gene id, position, codon, rfpcount. Each row corresponds to a single codon with an associated number of ribosome footprints.

```
GeneID,Position,Codon,rfpCount
YBR177C, 0, ATA, 8
```

```
YBR177C, 1, CGG, 1
YBR177C, 2, GTT, 8
YBR177C, 3, CGC, 1
```

2.4.4 Analyzing and Visualizing results

Parameter estimates

After we have completed the model fitting, we are interested in the results. AnaCoDa provides functions to obtain the posterior estimate for each parameter. For gene-set specific parameters or codon specific parameters we can use the function **getCSPEstimates**. Again we can specify for which mixture we would like the posterior estimate and how many samples should be used. **getCSPEstimates** has an optional argument filename which will cause the routine to write the result as a csv file instead of returning a **data.frame**.

```
cspMat <- getCSPEstimates(parameter = parameter, CSP="Mutation",
                          mixture = 1, samples = 1000)
head(cspMat)
# AA Codon Posterior 0.025% 0.975%
#1 A GCA -0.2435340 -0.2720696 -0.2165220
#2 A GCC 0.4235546 0.4049132 0.4420680
#3 A GCG 0.7004484 0.6648690 0.7351707
#4 C TGC 0.2016298 0.1679025 0.2387024
#5 D GAC 0.5775052 0.5618199 0.5936979
#6 E GAA -0.4524295 -0.4688044 -0.4356677
getCSPEstimates(parameter = parameter, filename = "mutation.csv",
                          CSP="Mutation", mixture = 1, samples = 1000)
```

To obtain posterior estimates for the gene specific parameters, we can use the function **getExpressionEstimatesForMixture**. In the case below we ask to get the gene specific parameters for all genes, and under the assumption each gene is assigned to mixture 1.

```
phiMat <- getExpressionEstimates(parameter = parameter,
                         gene.index = 1:length(genome),
                         samples = 1000)
head(phiMat)
# PHI log10.PHI Std.Error log10.Std.Error 0.025 0.975 log10.025 ...
#[1,] 0.2729446 -0.6188447 0.0001261525 2.362358e-04 0.07331819 ...
#[2,] 1.4221716 0.1498953 0.0001669425 5.194123e-05 1.09593642 ...
#[3,] 0.7459888 -0.1512764 0.0002313539 1.529267e-04 0.31559618 ...
#[4,] 0.6573082 -0.2030291 0.0001935466 1.400333e-04 0.31591233 ...
#[5,] 1.6316901 0.2098120 0.0001846631 4.986347e-05 1.28410352 ...
#[6,] 0.6179711 -0.2286806 0.0001744928 1.374863e-04 0.28478950 ...
```

However we can decide to only obtain certain gene parameters. in the first case we sample 100 random genes.

```
# sampling 100 genes at random
phiMat <- getExpressionEstimates(parameter = parameter,
                         gene.index = sample(1:length(genome), 100),
                         samples = 1000)
```

Furthermore, AnaCoDa allows to calculate the selection coefficient s for each codon and each gene. We can use the function **getSelectionCoefficients** to do so. Please note, that this function returns the $\log(sN_e)$.

getSelectionCoefficients returns a matrix with $\log(sN_e)$ relative to the most efficient synonymous codon.

```
selectionCoefficients <- getSelectionCoefficients(genome = genome,
                         parameter = parameter, samples = 1000)
head(selectionCoefficients)
# GCA GCC GCG GCT TGC TGT GAC GAT ...
#SAKL0A00132g -0.1630284 -0.008695144 -0.2097771 0 -0.1014373 ...
```

```
#SAKL0A00154g -0.8494558 -0.045305847 -1.0930388 0 -0.5285367 ...
#SAKL0A00176g -0.4455753 -0.023764823 -0.5733448 0 -0.2772397 ...
#SAKL0A00198g -0.3926068 -0.020939740 -0.5051875 0 -0.2442824 ...
#SAKL0A00220g -0.9746002 -0.051980440 -1.2540685 0 -0.6064022 ...
#SAKL0A00242g -0.3691110 -0.019686586 -0.4749542 0 -0.2296631 ...
```

We can compare these values to the weights from the codon adaptation index (CAI) citepsharp1987 or effective number of codons (N_c) (Wright, 1990) by using the functions **getCAIweights** and **getNcAA**.

```
caiWeights <- getCAIweights(referenceGenome = genome)
head(caiWeights)
# GCA GCC GCG GCT TGC TGT
#0.7251276 0.6282192 0.2497737 1.0000000 0.6222628 1.0000000
nc.per.aa <- getNcAA(genome = genome)
head(nc.per.aa)
# A C D E F G ...
#SAKL0A00132g 3.611111 1.000000 2.200000 2.142857 1.792453 ...
#SAKL0A00154g 1.843866 2.500000 2.035782 1.942505 1.986595 ...
#SAKL0A00176g 5.142857 NA 1.857143 1.652174 1.551724 3.122449 ...
#SAKL0A00198g 3.800000 NA 1.924779 1.913043 2.129032 4.136364 ...
#SAKL0A00220g 3.198529 1.666667 1.741573 1.756757 2.000000 ...
#SAKL0A00242g 4.500000 NA 2.095890 2.000000 1.408163 3.734043 ...
```

We can also compare the distribution of selection coefficients to the CAI values estimated from a reference set of genes. Figure 2.1, produced by the code below, shows that selection coefficients for the same codon can vary greatly between the genes.

```
selectionCoefficients <- getSelectionCoefficients(genome = genome,
                         parameter = parameter, samples = 1000)
s <- exp(selectionCoefficients)
```

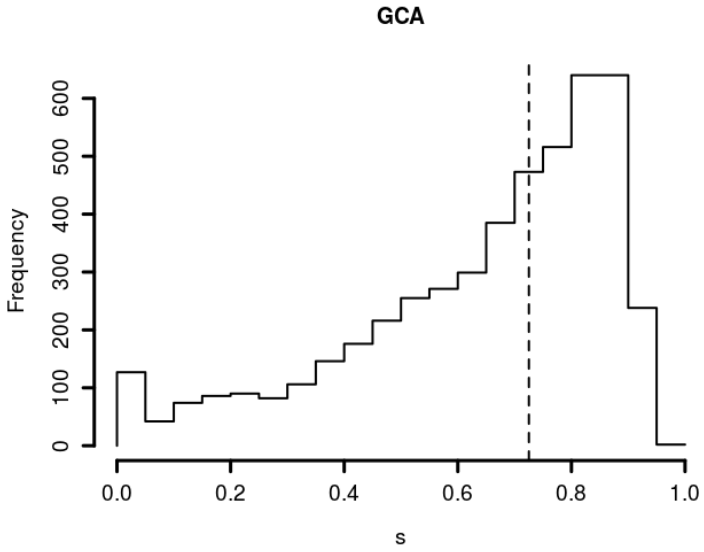


Figure 2.1: Distribution of s for codon GCA for amino acid alanine. Dashed line indicates the CAI weight for GCA. The comparison provides a more nuanced picture as we can see that the selection on GCA varies across the genome.

```
caiWeights <- getCAIweights(referenceGenome = ref.genome)
codonNames <- colnames(s)
h <- hist(s[, 1], plot = F)
plot(NULL, NULL, axes = F, xlim = c(0,1),
                          ylim = range(c(0,h$counts)),
                          xlab = "s", ylab = "Frequency",
                          main = codonNames[1], cex.lab = 1.2)
lines(x = h$breaks, y = c(0,h$counts), type = "S", lwd=2)
abline(v = cai.weights[1], lwd=2, lty=2)
axis(1, lwd = 3, cex.axis = 1.2)
axis(2, lwd = 3, cex.axis = 1.2)
```

Diagnostic Plots

A first step after every run should be to determine if the sampling routine has converged. To do that, AnaCoDa provides plotting routines to visualize all sampled parameter traces

from which the posterior sample is obtained (Figure 2.2). First we have to obtain the **trace** object stored within our **parameter** object. Now we can simply plot the **trace** object. The argument **what** specifies which type of parameter should be plotted. Here we plot the selection parameter $\Delta\eta$ of the ROC model. These parameters are mixture specific and one can decide which mixture set to visualize using the argument **mixture**.

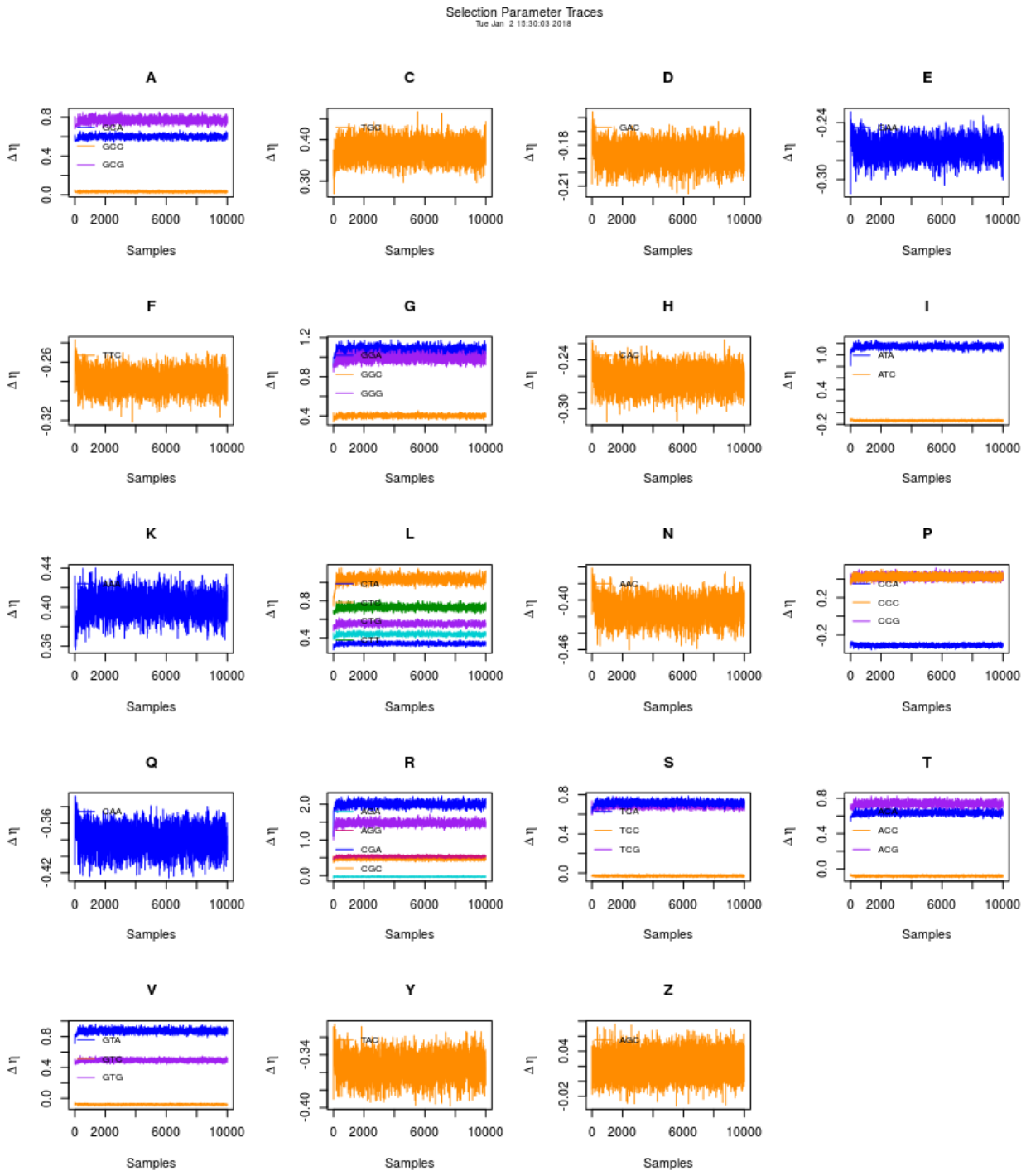


Figure 2.2: Trace plot showing the traces of all 40 codon specific selection parameters $\Delta\eta$ organized by amino acid.

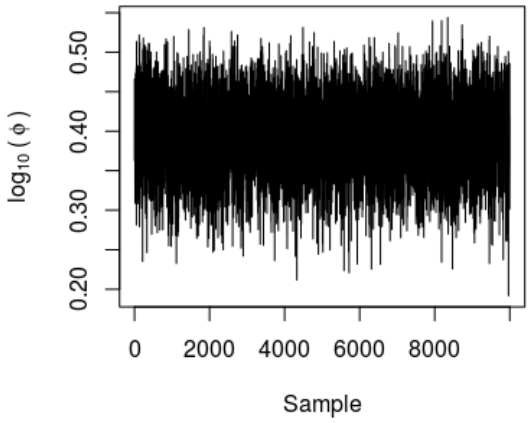


Figure 2.3: Trace plot showing the protein synthesis trace ϕ for gene 669.

```
trace <- getTrace(parameter)
plot(x = trace, what = "Selection", mixture = 1)
```

A special case is the plotting of traces of the protein synthesis rate ϕ (Figure 2.3). As the number of traces for the different ϕ traces is usually in the thousands, a **geneIndex** has to be passed to determine for which gene the trace should be plotted. This allows to inspect the trace of every gene under every mixture assignment.

```
trace <- parameter$getTraceObject()
plot(x = trace, what = "Expression", mixture = 1, geneIndex = 669)
```

We find the likelihood and posterior trace of the model fit in the **mcmc object**. The trace can be plotted by just passing the **mcmc** object to the **plot** routine. Again we can switch between log(*likelihood*) and log(*posterior*) using the argument **what**. The argument **zoom.window** is used to inspect a specified window in more detail. It defaults to the last 10 % of the trace. The log(*posterior*) displayed in the figure title is estimated over the **zoom.window** (Figure 2.4).

```
plot(mcmc, what = "LogPosterior", zoom.window = c(9000, 10000))
```

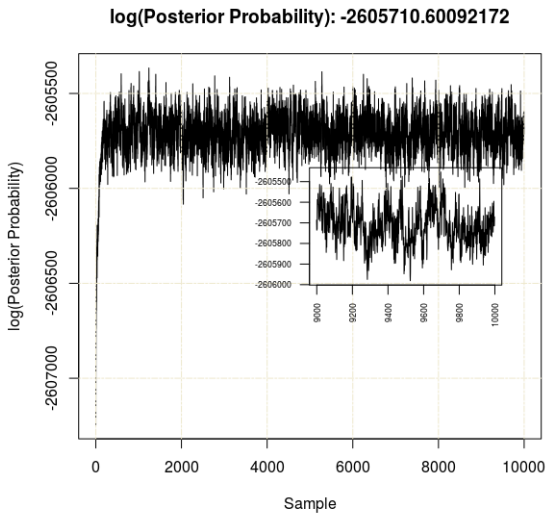


Figure 2.4: Trace plot showing the log(*posterior*) trace for the current model fit. Window inset shows the last 1.000 samples

Model visualization

We can visualize the results of the model fit by plotting the **model** object (Figure 2.5). For this we require the model and the **genome** object. We can adjust which mixture set we would like to visualize and how many samples should be used to obtain the posterior estimate for each parameter. For more details see Gilchrist *et al.* (2015).

```
# use the last 500 samples from mixture 1 for posterior estimate.
plot(x = model, genome = genome, samples = 500, mixture = 1)
```

As AnaCoDa is designed with the idea to allow gene-sets to have independent gene-set specific parameters, AnaCoDa also provides the option to compare different gene-sets by plotting the parameter object. Figure 2.6 allows us to compare the selection parameter estimated by ROC for seven yeast species. The code below illustrates how the figure is plotted.

```
# use the last 500 samples from mixture 1 for posterior estimate.
plot(parameter, what = "Selection", samples = 500)
```

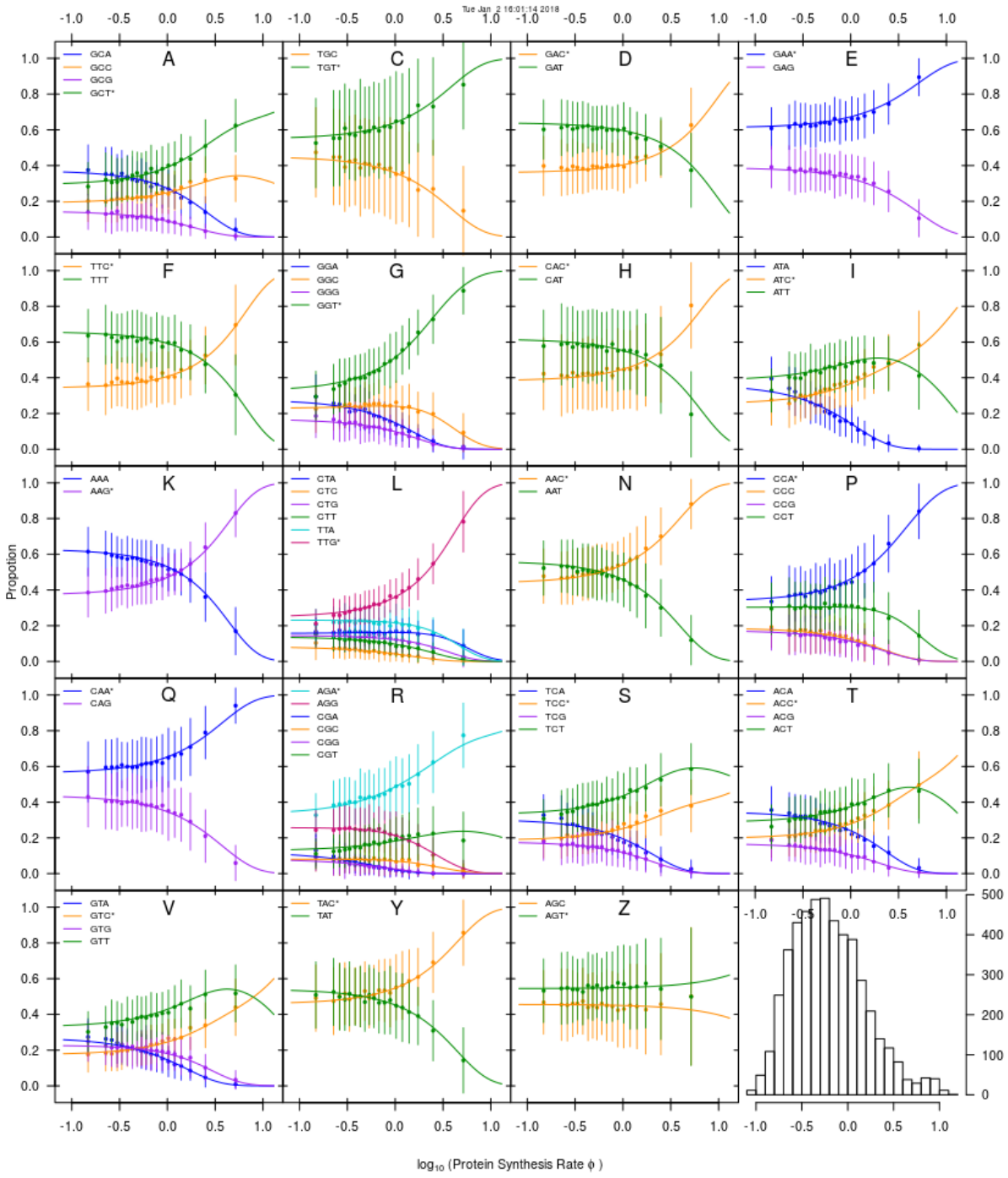


Figure 2.5: Fit of the ROC model for a random yeast. The solid line represent the model fit from the data, showing how synonymous codon frequencies change with gene expression. The points are the observed mean frequencies of a codon in that synthesis rate bin and the whisks indicate the standard deviation within the bin. The codon favored by selection is indicated by a "*". The bottom right panel shows how many genes are contained in each bin

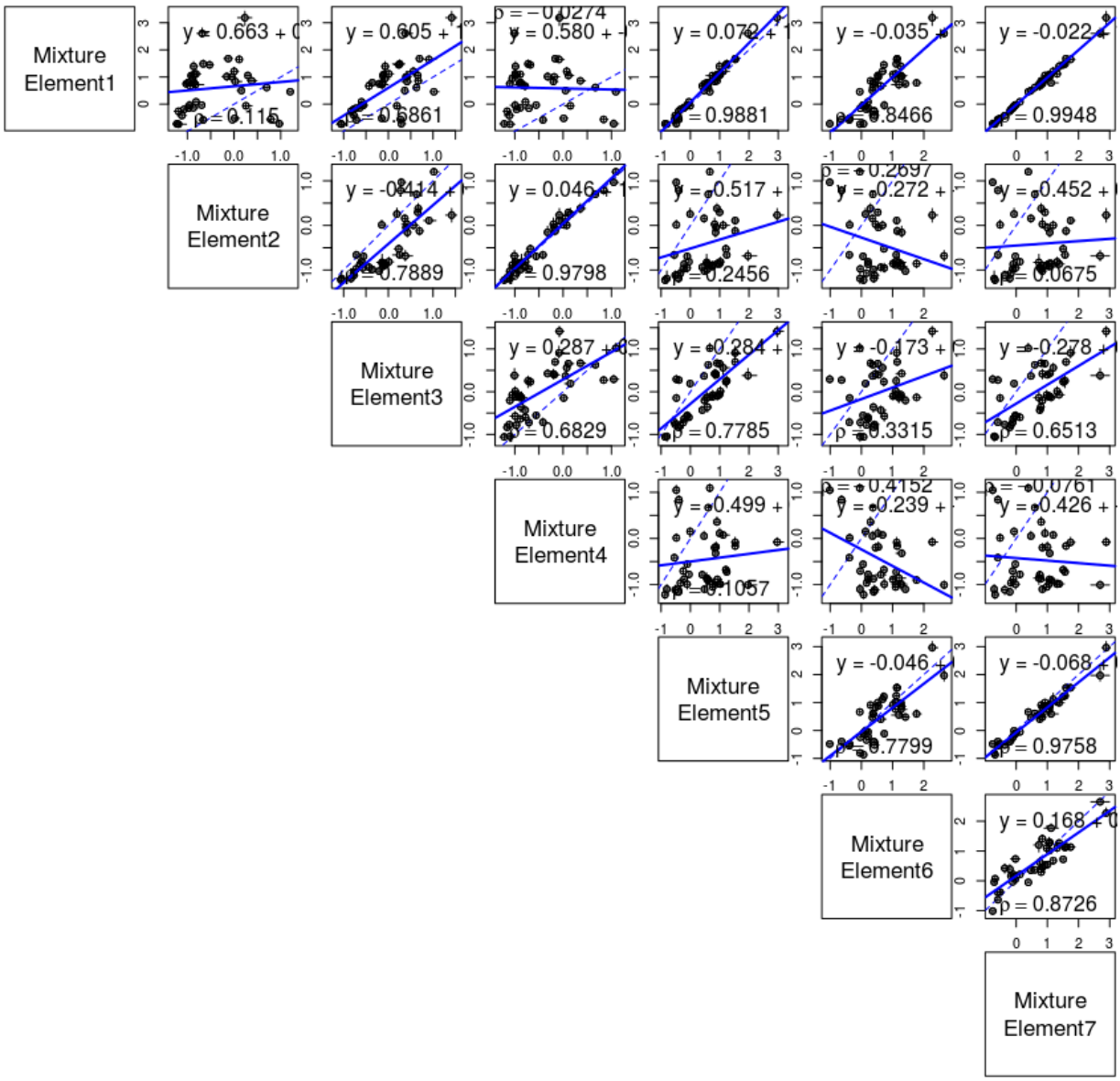


Figure 2.6: Comparison of the selection parameter of seven yeast species estimated with ROC-SEMPPR.

Chapter 3

Decomposing mutation and selection to identify mismatched codon usage

This chapter is a lightly revised version of a paper to be submitted to Genome Biology and Evolution and co-authored with Michael A. Gilchrist, Brian O'Meara, and Russel Zaretzki.

C. Landerer, B.C. O'Meara, R. Zaretzki, M.A. Gilchrist, Decomposing mutation and selection to identify mismatched codon usage

3.1 Abstract

For decades, codon usage has been used as a measure of adaptation for translational efficiency of a gene's coding sequence. These patterns of codon usage reflect both the selective and mutational environment in which the coding sequences evolved. Over this same period, gene transfer between lineages has become widely recognized as an important biological pheonmena. Nevertheless, most studies of codon usage implicitly assume that all genes within a genome evolved under the same selective and mutational environment, an assumption violated when introgression occurs. In order to better understand the effects of introgression on codon usage patterns and vice versa, we examine the patterns of codon usage in the yeast which has experienced a large introgression, *Lachancea kluyveri*. We quantify the effects of mutation bias and selection for translation efficiency on the codon usage pattern of the endogenous and introgressed exogenous genes using a Bayesian mixture model, ROC SEMPPR, which is built on mechanistic assumptions of protein synthesis and grounded in population genetics. We find substantial differences in codon usage between the endogenous and exogenous genes, and show that these differences can be largely attributed to a shift in mutation bias from A/T ending codons in the endogenous genes to C/G ending codons in the exogenous genes. Recognizing the two different signatures of mutation and selection bias improves our ability to predict protein synthesis rate by 17% and allowed us to accurately assess codon preferences. In addition, using our estimates of mutation and selection bias, we

to identify *Eremothecium gossypii* as the most likely source lineage, estimate the introgression occurred $\sim 6 \times 10^8$ generation ago, and estimate its historic and current genetic load. Together, our work illustrates the advantage of mechanistic, population genetic models like ROC SEMPPR and the quantitative estimates they provide when analyzing sequence data.

3.2 Introduction

Synonymous codon usage patterns varies within a genome and between taxa, reflecting differences in mutation bias, selection, and genetic drift. The signature of mutation bias is largely determined by the organism's internal or cellular environment, such as their DNA repair genes or UV exposure. While this mutation bias is an omnipresent evolutionary force, its impact can be obscured or even amplified by selection. The signature of selection on codon usage is also largely determined by an organism's cellular environment, such as its tRNA species, their copy number, and post-transcriptional modifications. The strength of selection on the codon usage of an individual gene is largely determined by its expression level which, in turn, is also largely determined by the organism's external environment. In general, the strength of selection on codon usage increases with its expression level (Gouy and Gautier, 1982; Ikemura, 1985; Bulmer, 1990), specifically its protein synthesis rate (Gilchrist, 2007). Thus as gene expression increases, codon usage shifts from a process dominated by mutation to a process dominated by selection. The overall efficacy of selection on codon usage is a function of the organism's effective population size N_e which, in turn, is largely determined by its external environment. By explicitly modeling the combined forces of mutation, selection, and drift, ROC SEMPPR allows us disentangle the evolutionary forces responsible for the patterns of codon usage bias (CUB) encoded in an species' genome (Gilchrist, 2007; Shah and Gilchrist, 2011a; Wallace *et al.*, 2013; Gilchrist *et al.*, 2015), should provide biologically meaningful information about the lineage's historical cellular and external environment.

Most studies implicitly assume that the CUB of a genome is shaped by a single cellular environment. As genes are horizontally transferred, introgress, or combined to form novel hybrid species, one would expect to see the influence of multiple cellular environments on a genomes codon usage pattern (Mdigue *et al.*, 1991; Lawrence and Ochman, 1997). Given that transferred genes are likely to be less adapted than endogenous genes to their new cellular environment, we expect a greater genetic load of transferred genes if donor and

recipient environment differ greatly in their selection bias, making such transfers less likely. More practically, if differences in codon usage of transferred genes are unaccounted for, they may distort parameter estimates. Such distortion could lead to the wrong codon preference for an amino acid, underestimate the variation in protein synthesis rate, or bias mutation estimates when analyzing a genome.

To illustrate these ideas, we analyze the CUB of the genome of *Lachancea kluyveri*, which is sister to all other Lachancea. The Lachancea clade diverged from the Saccharomyces clade, prior to its whole genome duplication $\sim$ 100 Mya ago (Marcet-Houben and Gabaldn, 2015; Beimforde *et al.*, 2014). Since that time, *L. kluyveri* has experienced a large introgression of exogenous genes found in all populations (Friedrich *et al.*, 2015). The introgression replaced the left arm of the C chromosome and displays a 13% higher GC content than the endogenous *L. kluyveri* genome (Payen *et al.*, 2009; Friedrich *et al.*, 2015). These characteristics make *L. kluyveri* an ideal model to study the effects of an introgressed cellular environment and the resulting mismatch in codon usage.

Using ROC SEMPPR, a Bayesian population genetics model based on a mechanistic description of ribosome movement along an mRNA, allows us to quantify the cellular environment in which genes have evolved by separately estimating the effects of mutation bias and selection bias on codon usage. ROC SEMPPR's resulting predictions of protein synthesis rates have been shown to be on par with laboratory measurements (Shah and Gilchrist, 2011a; Gilchrist *et al.*, 2015). In contrast to often used heurisitc approaches to study codon usage (Sharp and Li, 1987; dos Reis *et al.*, 2004), ROC SEMPPR explicitly incorpoates and distinguishes between mutation and selection effects on codon usage. We use ROC SEMPPR to independently describe two cellular environments reflected in the *L. kluyveri* genome; the signature of the current environment in the endogenous genes and the decaying signature of the exogenous environment in the introgressed genes. Our results indicate that the difference in GC content between endogenous and exogenous genes is mostly due to the differences in mutation bias of their ancestral environments. Accounting for these different

Table 3.1: Model selection of the two competing hypothesis. Reported are the log-likelihood, $\log(\mathcal{L})$, the number of parameters estimated n, AIC, and ΔAIC values.

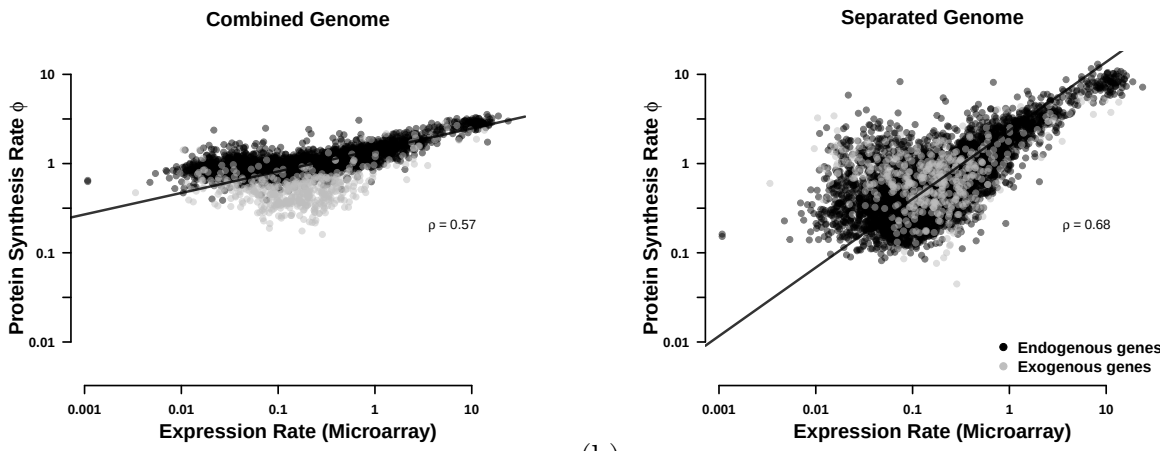
Hypothesis	$\log(\mathcal{L})$	n	AIC	ΔAIC
Separated	-2,612,397	5,402	5,235,598	0
Combined	-2,650,047	5,483	5,311,060	75,462

signatures of mutation bias and selection bias of the endogenous and exogenous sets of genes substantially improves our ability to predict present day protein synthesis rates. These endogenous and exogenous gene set specific estimates of mutation bias and selection bias, in turn allow us to address more refined questions of biological importance. For example, it allows us to identify *E. gossypii* as the most likely source of the introgressed genes out of the 38 yeast lineages with sequenced genomes, estimate the age of the introgression to be on the order of 0.2-1 Mya, estimate the genetic load of these genes, both at the time of introgression and now, as well as make predictions about how the CUB of the introgressed genes will evolve in the future.

3.3 Results

3.3.1 The Signatures of two Cellular Environments within *L. kluyveri*'s Genome

We used our software package AnaCoDa (Landerer *et al.*, 2018) to compare model fits of ROC SEMPPR to the entire *L. kluyveri* genome and its genome partitioned into two sets of 4,864 endogenous and 497 exogenous genes. AIC values strongly support the hypothesis that the *L. kluyveri* genome consists of genes with two different and distinct patterns of codon usage bias ($\Delta\text{AIC} = 75,462$; Table 3.1). We find additional support for this hypothesis when we compare our predictions of gene expression to empirically observed values. Specifically, the explanatory power between our predictions and observed values improved by $\sim 42\%$, from $R^2 = 0.33$ to 0.46 (Figure 3.1).



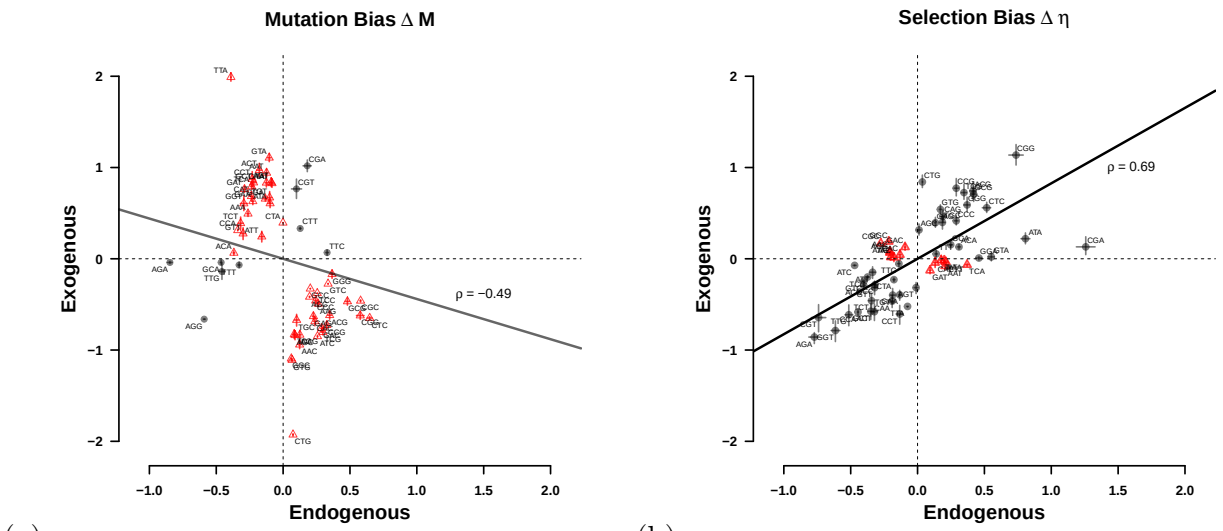
(a) (b)

Figure 3.1: Comparison of predicted protein synthesis rate ϕ to microarray data from Tsankov *et al.* (2010) for (a) the combined genome and (b) the separated endogenous and exogenous genes. Endogenous genes are displayed in black and exogenous genes in gray. Black line indicates type II regression line (Sokal and Rohlf, 1981).

3.3.2 Comparing Differences in the Endogenous and Exogenous Codon Usage

To better understand the differences in the endogenous and exogenous cellular environments, we compared our parameter estimates of mutation bias ΔM and selection $\Delta\eta$ for the two sets of genes. Our estimates of ΔM for the endogenous and exogenous genes were negatively correlated ($\rho = -0.49$), indicating weak concordance of $\sim 5\%$ between the two mutation environments (Figure 3.2). For example, the endogenous genes show a mutational preference for A and T ending codons in $\sim 95\%$ of the codon families. In contrast, the exogenous genes display an equally consistent mutational preference towards C and G ending codons (Table 3.2). As a result, only the two codon amino acid Phenylalanine (Phe, F) shares the same rank order across the endogenous and exogenous ΔM estimates.

In contrast, our estimates of $\Delta\eta$ for the endogenous and exogenous genes were positively correlated ($\rho = 0.69$) and showing concordance of $\sim 53\%$ between the two selection environments (Figure 3.2). ROC SEMPPR constraints $E[\phi] = 1$, allowing us to interpret $\Delta\eta$ as selection on codon usage of the average gene with $\phi = 1$ and gives us the ability to



(a) (b)

Figure 3.2: Comparison of (a) mutation bias ΔM and (b) selection bias $\Delta\eta$ parameters for endogenous and exogenous genes. Estimates are relative to the mean for each codon family. Black dots indicate ΔM or $\Delta\eta$ parameters with the same sign for the endogenous and exogenous genes, red dots indicate parameters with different signs. Black line shows the type II regression line (Sokal and Rohlf, 1981). Dashed lines mark quadrants.

compare the efficacy of selection sN_e across genomes. We find that the strength of selection within each codon family differs between sets of genes. Overall, the endogenous genes only show a selection preference for C and G ending codons in $\sim$ 58% of the codon families. In contrast, the exogenous genes display a strong preference for A and T ending codons in $\sim$ 89% of the codon families.

The difference in codon usage between endogenous and exogenous genes is striking. As a result, our estimates of the optimal codon differ in nine cases between endogenous and exogenous genes (Table 3.3). Fits to the complete *L. kluyveri* genome reveal that the relatively small exogenous gene set ($\sim$ 10% of genes) has a disproportional effect on the model fit. We find that the complete *L. kluyveri* genome is estimated to share the mutational preference with the exogenous genes in $\sim$ 78% of the 19 codon families that are discordant between the endogenous and exogenous genes. In two cases, Isoleucine (Ile, I) and Arginine (Arg, R), the strong discordance in mutation preference results in an estimated

codon preference in the complete *L. kluyveri* genome that differs from both the endogenous, and the exogenous genes.

The effect of the small exogenous gene set on the fit to the complete *L. kluyveri* genome is smaller in our estimates of selection bias $\Delta\eta$ than ΔM, but still large. We find that the complete *L. kluyveri* genome is estimated to share the selection preference with the exogenous genes in $\sim 60\%$ of codon families that show discordance between endogenous and exogenous genes. These results clearly show that it is important to recognize the difference in endogenous and exogenous genes and treat these genes as separate sets to avoid the inference of incorrect synonymous codon preferences and better predict protein synthesis.

3.3.3 Determining Source of Exogenous Genes

We combined our estimates of mutation bias ΔM and selection bias $\Delta\eta$ with synteny information and searched for potential source lineages of the introgressed exogenous region. We examined 38 yeast lineages (Table 3.4) of which two (*Eremothecium gossypii* and *Candida dubliniensis*) showed a strong positive correlation in codon usage (Figure 3.3). The endogenous *L. kluyveri* genome exhibits codon usage very similar to most yeast lineages examined, indicating little variation in codon usage among the examined yeasts (Figure 3.5). Four lineages show a positive correlation for ΔM and $\Delta\eta$ with the exogenous genes and have a weak to moderate positive correlation in selection bias with the endogenous genes; but, like the exogenous genes, tend to have a negative correlation in ΔM with the endogenous genes.

Comparing synteny between the exogenous genes, which are restricted to the left arm of chromosome C, and *E. gossypii* and *C. dubliniensis* as well as closely related yeast species we find that *E. gossypii* displays the highest synteny (Figures 3.7 & 3.8). *C. dubliniensis*, even though it displays similar codon usage does not show synteny with the exogenous region. Furthermore, the synteny relationship between the exogenous region and other yeasts appears to be limited to the Saccharomycetacease clade (Figure 3.8). Given these results, we conclude

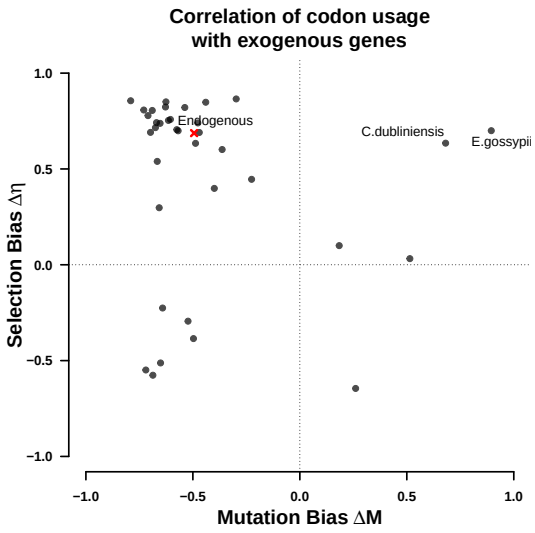


Figure 3.3: Correlation coefficients of ΔM and $\Delta\eta$ of the exogenous genes with 38 examined yeast lineages. Dots indicate the correlation of ΔM and $\Delta\eta$ of the lineages with the endogenous and exogenous parameter estimates. All regressions were performed using a type II regression (Sokal and Rohlf, 1981).

that of the 38 examined yeast lineages the *E. gossypii* lineage is the most likely source of the introgressed exogenous genes.

3.3.4 Estimating Introgression Age

We modeled the change in codon frequency as a model of exponential decay, we estimated the age of the introgression assuming that *E. gossypii* still represents the mutation bias of its ancestral source lineage at the time of the introgression and a constant mutation rate. We infer the age of the introgression to be on the order of $6.2 \pm 1.2 \times 10^8$ generations. Assuming *L. kluyveri* experiences between one and eight generations per day, we estimate the introgression to have occurred between $212,000$ to $1,700,000$ years ago. Our estimate places the time of the introgression earlier than previously assumed (Friedrich *et al.*, 2015).

Using the same approach, we also estimated the persistence of the signal of the exogenous cellular environment. We assume that differences in mutation bias will decay more slowly than differences in selection bias to be able to utilize our bias free estimates of ΔM. We

predict that the ΔM signal of the source cellular environment will have decayed to be within one percent of the *L. kluyveri* environment in $\sim 5.4 \pm 0.2 \times 10^9$ generations, or between $1,800,000$ and $15,000,000$ years. Together, these results indicate that the mutation signature of the exogenous genes will persist for a very long time.

3.3.5 Genetic Load due to Mismatching Codon Usage of the Exogenous Genes

We define genetic load as the difference between the fitness of an expected, replaced endogenous gene and the exogenous gene, $s \propto \phi\Delta\eta$ due to the mismatch in codon usage parameters (See Methods for details). Estimates of selection bias for the exogenous genes show that, while well correlated with the endogenous genes, only nine amino acids share the same optimal codon. Exogenous genes are, therefore, expected to represent a significant reduction in fitness, or genetic load for *L. kluyveri* due to this mismatch in codon usage. As the introgression occurred before the diversification of *L. kluyveri* and has fixed throughout all populations (Friedrich *et al.*, 2015), we can not observe the original endogenous sequences that have been replaced by the introgression. Using our estimates of ΔM and $\Delta\eta$ from the endogenous genes and assuming hat the current exogenous amino acid composition of genes is representative of the replaced endogenous genes, we estimate the genetic load of the exogenous genes at the time of introgression (Figure 3.4a) and currently (Figure 3.4b). We find that the genetic load due to mismatched codon usage was -0.0008 at the time of the introgression and still represents a genetic load of -0.0003 today.

In order to account for differences in the efficacy of selection on codon usage between the donor lineage and *L. kluyveri* using a linear scaling factor κ (See Methods for details). We predict that a small number of low expression genes ($\phi < 1$) were weakly exapted at the time of the introgression (Figure 3.4a). High expression genes ($\phi > 1$) are predicted to have carried the largest genetic load in the novel cellular environment. These highly expressed

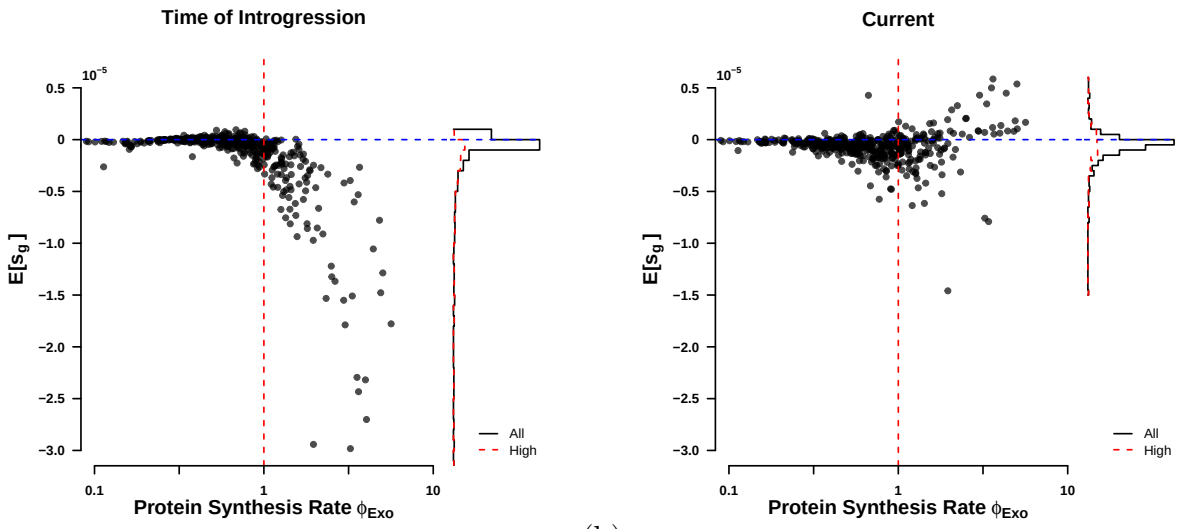


Figure 3.4: Genetic load $s = \Delta\eta\phi$ (a) at the time of introgression ($\kappa = 5$), and (b) currently ($\kappa = 1$).

genes are inferred to have the greatest degree of adaptation since the time of the introgression to the *L. kluyveri* cellular environment (Figures 3.4a & 3.10).

3.4 Discussion

In order to study the evolutionary effects of an introgression, we used ROC SEMPPR, a mechanistic model of ribosome movement along an mRNA. Our parameter estimates indicate that the *L. kluyveri* genome contains distinct signatures of mutation and selection bias from both an endogenous and exogenous cellular environment. By fitting ROC SEMPPR separately to *L. kluyveri*'s endogenous and exogenous sets of genes we generate a quantitative description of their signatures of mutation bias and natural selection for efficient protein translation. Our results indicate that the difference in GC content between endogenous and exogenous genes is mostly due to differences in mutation bias, but we also show that the strength and rank order of selection within a codon family differ between endogenous and exogenous cellular environments. Even though the exogenous genes make up only $\sim 10\%$ of

the *L. kluyveri* genome, when we fail to recognize these differences our estimates of ΔM and $\Delta\eta$ deviate substantial from their actual values (Figure 3.6). While this sensitivity of our parameters to a second cellular environment may be surprising, it highlights the importance of recognizing different cellular environments reflected by a genome. Furthermore, our results indicate that we can attribute the increased GC content in the exogenous genes mostly to differences in mutation bias favoring G/C ending codons rather than selection.

The separation of the endogenous and exogenous genes improves our estimates of protein synthesis rate ϕ by 42% relative to the full genome estimate ($R^2 = 0.32$ vs. 0.46, respectively). Furthermore, failing to separately analyze the endogenous and exogenous genes results in an unrealistically small amount of intergenic variation in ϕ (compare Figure 3.1a & b). This behavior is due, in part, to constraining $E[\phi] = 1$ which allows us to compare the efficacy of selection sN_e across genomes. Extremely small variances in the ϕ values estimated by ROC SEMPPR could indicate that a genome contains the signature of multiple cellular environments.

The mutation and selection bias parameters ΔM and $\Delta\eta$ of the introgressed exogenous genes contain information, albeit decaying, about its previous cellular environment. We, therefore, utilize ΔM and $\Delta\eta$ to identify potential source lineages. The *E. gossypii* and *C. dubliniensis* lineages stand out from the other 36 yeast lineages in that the correlation coefficients between their ΔM and $\Delta\eta$ parameters and those of the exogenous genes are > 0.5 (Figure 3.2). In terms of gene order, we found that synteny with the exogenous genes is limited to the Saccharomycetaceae clade, which *C. dubliniensis* is outside of. Overall, the synteny coverage extends along the whole exogenous regions with the exception of the 3' and 5' ends of the exogenous region (Figure 3.8b). Further, of the 38 species examined, *E. gossypii* is the only genome with a GC content $> 50\%$, making it most similar to the exogenous genes. Thus, only the *E. gossypii* genome displays strong correlations in ΔM and $\Delta\eta$, synteny, and similar GC content with the exogenous genes.

With *E. gossypii* identified as potential source lineage of the introgressed region, we inferred the time since the introgression occurred using our estimates of mutation bias ΔM. Our ΔM estimates are well suited for this task as they are free of the influence of selection and unbiased by N_e and other scaling terms, which is in contrast to our estimates of $\Delta\eta$ (Gilchrist *et al.*, 2015). Our estimated age of the introgression of $6.2 \pm 1.2 \times 10^8$ generations is ~ 10 times longer time than a previous minimum estimate by Friedrich *et al.* (2015) of 5.6×10^7 generations. Our estimate assumes that the current *E. gossypii* and *L. kluyveri* cellular environment reflect their ancestral states at the time of the introgression. If the ancestral mutation environments were more similar (dissimilar) at the time of the introgression than now our result is an overestimate (underestimate).

In order to estimate the introgression's genetic load due to codon mismatch, we had to make three key assumptions: 1) at the time of introgression the amino acid sequences of the endogenous genes and exogenous genes where highly similar, 2) the current *L. kluyveri* cellular environment is reflective of the cellular environment at the time of the introgression, and 3) the *E. gossypii* cellular environment reflects its ancestral environment at the time of the introgression. In general due to their very nature, low expression genes contribute little to the genetic load. Indeed, $\sim 30\%$ of low expression exogenous genes ($\phi < 1$) appeared to be exapted at the time of the introgression. These exapted genes are likely due to the mutation bias in the endogenous genes matching the selection bias in the exogenous genes for G/C ending codons. In contrast, highly expressed genes are predicted to have imposed a large genetic load. Many of these genes appear to still represent a significant genetic load. Overall, our estimates of codon mismatch genetic load, therefore, suggest strong selection against the introgression.

It is hard to contextualize the probability of this introgression being fixed as we are not aware of any estimates of the frequency at which such large scale introgressions of genes occur. A related example of a large scale merger of genomic material can be found in *S. pastorianus*, which is currently believed to be a hybrid of *S. cerevisiae* and *S. eubayanus*

lineages, (Baker *et al.*, 2015). Unlike with *L. kluyveri* and *E. gossypii*, the progenitor lineages of *S. pastorianus* have similar codon usage parameters. The correlation between ΔM and $\Delta\eta$ for these two lineages are $\rho = 0.83$ and 0.98 (data not shown). These similarities in ΔM and $\Delta\eta$ parameters suggest that the genetic load for *S. pastorianus* due to codon usage mismatch is small relative to the exongenous genes considered here. The large genetic load of the exogenous genes due to codon mismatch at the time of the introgression would seem to indicate that the fixation of the introgression was either a fluke event or the codon mismatch genetic load was countered by one or more highly advantageous loci within the introgression.

Under the first scenario, our best estimate of the selection coefficient against the introgression based on expected codon mismatch at that time is $s = -0.0008$ and an effective population size N_e on the order of 10^8 (Wagner, 2005) yields an approximate fixation probability of $(1 - \exp[-s])/(1 - \exp[2 - sN_e]) \approx 10^{-6950}$ (Sella and Hirsh, 2005). Even though *L. kluyveri* diverged from the rest of the Lachancea clade around 85 Mya (Kensche *et al.*, 2008; Marcet-Houben and Gabaldn, 2015), if we assume 1 to 8 generations/day, which implies 10^{10} to 10^{11} generations since the time of divergence, one round of meiosis for every 1000 rounds of mitosis based on *S. paradoxus* (Tsai *et al.*, 2008), and $N_e \approx 10^8$ there were only 10^{15} to 10^{16} opportunities for such an introgression to have occurred and fixed. Clearly, unless there was a severe bottleneck with $N_e < 1/|s| \approx 1,250$ around the time of introgression, which conceivably could have been triggered by a speciation event, this scenario seems very unlikely.

In the second scenario, where we assume the introgression contained advantageous loci, one may wonder why recombination events did not limit the introgression to only the adaptive loci. Payen *et al.* (2009) found that the exogenous region has a lower rate of recombination, presumably due to the dissimilarity in GC content and/or a lower than average sequence homology between the exogenous region and the one it replaced. Compatible with this explanation is the possibility of several highly advantageous loci distributed across the

region which then drove a rapid selective sweep and/or the population through a bottleneck speciation process A careful analysis of intra-specific genetic variation within the endogenous and exongenous regions could provide help us distinguish between these various scenarios.

Overall, our results show the usefulness of the separation of mutation bias and selection bias and the importance of recognizing the presence of multiple cellular environments in the study of codon usage. We also illustrate how a mechanistic model like ROC SEMPPR and the quantitative estimates it provides can be used for more sophisticated hypothesis testing in the future. In contrast to other approaches used to study codon usage like CAI (Sharp and Li, 1987) or tAI (dos Reis *et al.*, 2004), ROC SEMPPR incorporates the effects of mutation bias and amino acid composition explicitly Cope *et al.* (2018). We highlight potential issues when estimating codon preferences, as estimates can be biased by the signature of a second, historical cellular environment. In addition, we show how quantitative estimates of mutation bias and selection relative to drift can be obtained from codon data and used to infer the fitness cost of an introgression as well as its history and potential future.

3.5 Materials and Methods

3.5.1 Separating Endogenous and Exogenous Genes

A GC-rich region was identified by Payen *et al.* (2009) in the *L. kluyveri* genome extending from position 1 to 989,693 of chromosome C. This region was later identified as an introgression by Friedrich *et al.* (2015). We obtained the *L. kluyveri* genome from SGD Project http://www.yeastgenome.org/download-data/ (on 09-27-2014) and the annotation for *L. kluyveri* NRRL Y-12651 (assembly ASM14922v1) from NCBI (on 12-09-2014). We assigned 457 genes located on chromosome C with a location within the $\sim 1Mb$ window to the exogenous gene set. All other 4864 genes of the *L. kluyveri* genome were assigned to the exogenous genes. All genes could be uniquely assigned to one or the other gene set.

3.5.2 Model Fitting with ROC SEMPPR

ROC SEMPPR was fitted to each genome using AnaCoDa (0.1.1) (Landerer *et al.*, 2018) and R (3.4.1) (R Core Team, 2015). ROC SEMPPR was run from multiple starting values for at least 250,000 iterations, only every 50th step was collected as a sample to reduce autocorrelation. After manual inspection to verify that the MCMC had converged, parameter posterior means were estimated from the last 500 samples.

3.5.3 Comparing Codon Specific Parameter Estimates

Choice of reference codon does reorganize codon families coding for an amino acid relative to each other, therefore all parameter estimates are relative to the mean for each codon family.

$$\Delta M^{c}_{i,a} = \Delta M_{i,a} - \overline{\Delta M_a} \tag{3.1}$$

$$\Delta \eta^{c}_{i,a} = \Delta \eta_{i,a} - \overline{\Delta \eta_a} \tag{3.2}$$

Comparison of codon specific parameters (ΔM and $\Delta \eta = 2N_e q(\eta_i - \eta_j)$) was performed using the function lmodel2 in the R package lmodel2 (1.7.3) (Legendre, 2018) and R version 3.4.1 (R Core Team, 2015). The parameter $\Delta \eta$ can be interpreted as the difference in fitness between codon i and j for the average gene with $\phi = 1$ scaled by the effective population size N_e, and the selective cost of an ATP q (Gilchrist, 2007; Gilchrist *et al.*, 2015). Type II regression was performed with re-centered parameter estimates, accounting for noise in dependent and independent variable (Sokal and Rohlf, 1981).

3.5.4 Synteny Comparison

We obtained complete genome sequences from NCBI (on: 02-05-2017). Genomes were aligned and checked for synteny using SyMAP (4.2) with default settings (Soderlund

et al., 2006, 2011). We assess synteny as percentage coverage of the exogenous gene region (Figure 3.8b).

3.5.5 Estimating Age of Introgression

We modeled the change in codon frequency over time using an exponential model for all two codon amino acids, and describing the change in codon c_1 as

$$\frac{dc_1}{dt} = -\mu_{1,2}c_1 - \mu_{2,1}(1 - c_1) \tag{3.3}$$

where $\mu_{i,j}$ is the rate at which codon i mutates to codon j and c_1 is the frequency of the reference codon. Our estimates of ΔM_{endo} can be used to calculate the steady state of equation 3.3.

$$\frac{\mu_{2,1}}{\mu_{1,2} + \mu_{2,1}} = \frac{1}{1 + \exp[\Delta M_{\text{endo}}]} \tag{3.4}$$

Solving for $\mu_{1,2}$ gives us $\mu_{1,2} = \Delta M_{\text{endo}} \exp[\mu_{2,1}]$ which allows us to rewrite and solve equation 3.3 as

$$c_1(t) = \frac{\exp[-t(1 + \Delta M_{\text{endo}})\mu_{2,1}] \exp[t(1 + \Delta M_{\text{endo}})\mu_{2,1}] + (1 + \Delta M_{\text{endo}})K}{1 + \Delta M_{\text{endo}}} \tag{3.5}$$

where K is

$$K = c_1(0) - \frac{1}{1 + \Delta M_{\text{endo}}} \tag{3.6}$$

Equation 3.5 was solved with a mutation rate $m_{2,1}$ of 3.8×10^{-10} per nucleotide per generation (Lang and Murray, 2008). Initial codon frequencies $c_1(0)$ for each codon family where taken from our mutation parameter estimates for *E. gossypii* ΔM_{gos}. Current codon frequencies for each codon family where taken from our estimates of ΔM from the exogenous genes. Mathematica (11.3) (Wolfram Research Inc., 2017) was used to calculate the time t_{intro} it takes for the initial codon frequencies $c_1(0)$ for each codon family to equal the current exogenous codon frequencies. The same equation was used to determine the time

t_{decay} at which the signal of the exogenous cellular environment has decayed to within 1% of the endogenous environment.

3.5.6 Estimating Genetic Load

To estimate the genetic load due to mismatched codon usage, we made three key assumptions. First, we assumed that the current exogenous amino acid sequence of a gene is representative of its ancestral state and the replaced endogenous gene it replaced. Second, we assume that the currently observed cellular environment of *E. gossypii* reflects the cellular environment that the exogenous genes experienced before transfer to *L. kluyveri*. Lastly, we assume that the difference in the efficacy of selection between the cellular environments due to differences in either effective population size N_e or the selective cost of an ATP q of the source lineage and *L. kluyveri* can be expressed as a scaling constant and that protein synthesis rate ϕ has not changed between the replaced endogenous and the introgressed exogenous genes. Using estimates for $N_e = 1.36 \times 10^7$ (Wagner, 2005) for *Saccharomyces paradoxus* we scale our estimates of $\Delta\eta$ and define $\Delta\eta' = \frac{\Delta\eta}{N_e}$.

We scale the difference in the efficacy of selection on codon usage between the donor lineage and *L. kluyveri* using a linear scaling factor κ. As $\Delta\eta$ is defined as $\Delta\eta = 2N_e q(\eta_i - \eta_j)$, we can not distinguish if κ is a scaling on protein synthesis rate ϕ, effective population size N_e, or the selective cost of an ATP q (Gilchrist, 2007; Gilchrist *et al.*, 2015). We calculated the genetic load each gene represents due to its mismatched codon usage assuming additive fitness effects as

$$s_g = \sum_{i=1}^{n_g} -\kappa\phi_g\Delta\eta_i' \tag{3.7}$$

where s_g is the overall strength of selection for translational efficiency on gene, g in the exogenous gene set, κ is a constant, scaling the efficacy of selection between the endogenous and exogenous cellular environments, n_g is length of the protein, ϕ_g is the estimated protein synthesis rate of the gene in the endogenous environment, and $\Delta\eta_i'$, is the $\Delta\eta$' for the codon

at position i. As stated previously, our $\Delta\eta$ are relative to the mean of the codon family. We find that the genetic load of the introgressed genes is minimized at $\kappa \sim 5$ (Figure 3.9b). Thus, we expect a five fold difference in the efficacy of selection between *L. kluyveri* and *E. gossypii*, either due to differences in either protein synthesis rate ϕ, effective population size N_e, or the selective cost of an ATP q. Therefore, we set $\kappa = 1$ if we calculate the s_g for the endogenous and the current exogenous genes, and $\kappa = 5$ for s_g for the genetic load at the time of introgression.

Since we are unable to observe codon counts for the replaced endogenous genes and for the exogenous genes at the time of introgression, we calculate expected codon counts

$$E[n_{g,i}] = \frac{\exp[-\Delta M_i - \Delta\eta_i\phi_g]}{\sum_j^C \exp[-\Delta M_j - \Delta\eta_j\phi_g]} \times m_{a_i} \tag{3.8}$$

m_{a_i} is the number of occurrences of amino acid a that codon i codes for. We report the genetic load due to mismatched codon usage of the introgression as $E[s_g] = s_{\text{intro},g} - s_{\text{endo},g}$ where $s_{\text{intro},g}$ is the genetic load of an introgressed gene g either at the time of the introgression or presently.

3.6 Acknowledgments

This work was supported in part by NSF Awards MCB-1120370 (MAG and RZ) and DEB-1355033 (BCO, MAG, and RZ) with additional support from The University of Tennessee Knoxville. CL received support as a Graduate Student Fellow at the National Institute for Mathematical and Biological Synthesis, an Institute sponsored by the National Science Foundation through NSF Award DBI-1300426, with additional support from UTK. The authors would like to thank Alexander Cope for helpful criticisms and suggestions for this work.

3.7 Appendix: Supplementary Material

Table 3.2: Synonymous codon preference in the various data sets based on our estimates of ΔM

Amino Acid	*E. gossypii*	Endogenous	Exogenous	*L. kluyveri*
Ala A	GCG	GCA	GCG	GCG
Cys C	TGC	TGT	TGC	TGC
Asp D	GAC	GAT	GAC	GAC
Glu E	GAG	GAA	GAG	GAG
Phe F	TTC	TTT	TTT	TTT
Gly G	GGC	GGT	GGC	GGC
His H	CAC	CAT	CAC	CAC
Ile I	ATC	ATT	ATC	ATA
Lys K	AAG	AAA	AAG	AAA
Leu L	CTG	TTG	CTG	CTG
Asn N	AAC	AAT	AAC	AAT
Pro P	CCG	CCA	CCG	CCG
Gln Q	CAG	CAA	CAG	CAG
Arg R	CGC	AGA	AGG	CGG
Ser_4 S	TCG	TCT	TCG	TCG
Thr T	ACG	ACA	ACG	ACG
Val V	GTG	GTT	GTG	GTG
Tyr Y	TAC	TAT	TAC	TAC
Ser_2 Z	AGC	AGT	AGC	AGC

Table 3.3: Synonymous codon preference in the various data sets based on our estimates of $\Delta\eta$

Amino Acid	*E. gossypii*	Endogenous	Exogenous	*L. kluyveri*
Ala A	GCT	GCT	GCT	GCT
Cys C	TGT	TGT	TGT	TGT
Asp D	GAT	GAC	GAT	GAT
Glu E	GAA	GAA	GAA	GAA
Phe F	TTT	TTC	TTC	TTC
Gly G	GGA	GGT	GGT	GGT
His H	CAT	CAC	CAT	CAT
Ile I	ATA	ATC	ATT	ATT
Lys K	AAA	AAG	AAA	AAG
Leu L	TTA	TTG	TTG	TTG
Asn N	AAT	AAC	AAT	AAC
Pro P	CCA	CCA	CCT	CCA
Gln Q	CAA	CAA	CAA	CAA
Arg R	AGA	AGA	AGA	AGA
Ser_4 S	TCA	TCC	TCT	TCT
Thr T	ACT	ACC	ACT	ACT
Val V	GTT	GTC	GTT	GTT
Tyr Y	TAT	TAC	TAT	TAC
Ser_2 Z	AGT	AGT	AGT	AGT

Table 3.4: Overview of yeast lineages used in this study.

Taxon	Abbreviation	NCBI taxonomic ID	Codon Table	% GC
Candida albicans	Calb	5476	12	34
Saccharomyces bayanus	Sbay	4931	1	40
Trichophyton benhamiae	Tben	63400	1	49
Tetrapisispora blattae	Tbla	1071379	1	32
Saccharomyces castellii	Scas	27288	1	37
Saccharomyces cerevisiae	Scer	4932	1	38
Eremothecium cymbalariae	Ecym	45285	1	40
Torulaspora delbrueckii	Tdel	4950	1	42
Candida dubliniensis	Cdub	42374	12	33
Lodderomyces elongisporus	Lelo	36914	1	37
Saccharomyces eubayanus	Seub	1080349	1	40
Debaryomyces fabryi	Dfab	58627	1	36
Candida glabrata	Cgla	5478	1	39
Eremothecium gossypii	Egos	33169	1	52
Meyerozyma guilliermondii	Mgui	4929	12	44
Debaryomyces hansenii	Dhan	4959	12	36
Lachancea kluyveri	Lku	4934	1	40/53
Saccharomyces kudriavzevii	Skud	114524	1	41
Kluyveromyces lactis	Klac	28985	1	39
Lachancea lanzarotensis	Llan	1245769	1	44
Yarrowia lipolytica	Ylip	4952	1	49
Clavispora lusitaniae	Clus	36911	12	45
Kluyveromyces marxianus	Kmar	4911	1	40
Saccharomyces mikatae	Smik	114525	1	38
Sphaerulina musiva	Smus	85929	1	51
Kazachstania naganishii	Knag	588726	1	46
Saccharomyces paradoxus	Spar	27291	1	38
Candida parapsilosis	Cpar	5480	12	38
Spathaspora passalidarum	Spas	340170	12	38
Tetrapisispora phaffii	Tpha	113608	1	34
Vanderwaltozyma polyspora	Vpol	36033	1	33
Lachancea quebecensis	Lque	1654605	1	47
Zygosaccharomyces rouxii	Zrou	4956	1	40
Scheffersomyces stipitis	Ssti	4924	12	41
Lachancea thermotolerans	Lthe	381046	1	47
Candida tropicalis	Ctro	5482	12	33
Lachancea waltii	Lwal	4914	1	44
Cladophialophora yegresii	Cyeg	470704	1	54

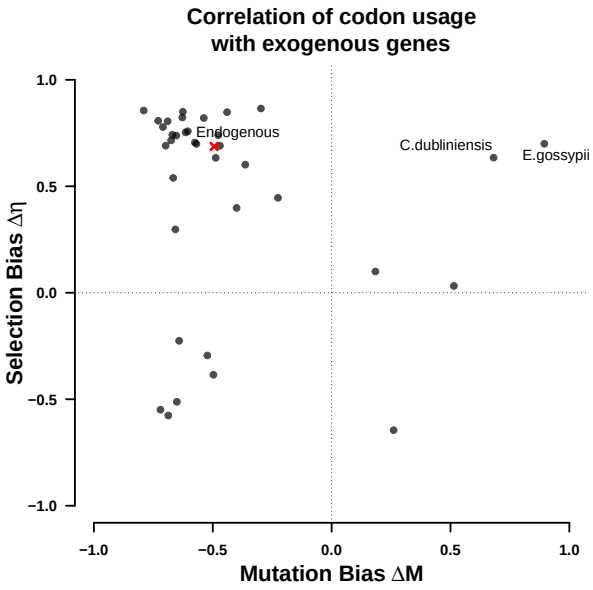


Figure 3.5: Correlation coefficient of ΔM and $\Delta\eta$ of the endogenous genes with 38 examined yeast lineages. Dots indicate the correlation of ΔM and $\Delta\eta$ of the lineages with the endogenous and exogenous parameter estimates. All regressions were performed using a type II regression line (Sokal and Rohlf, 1981).

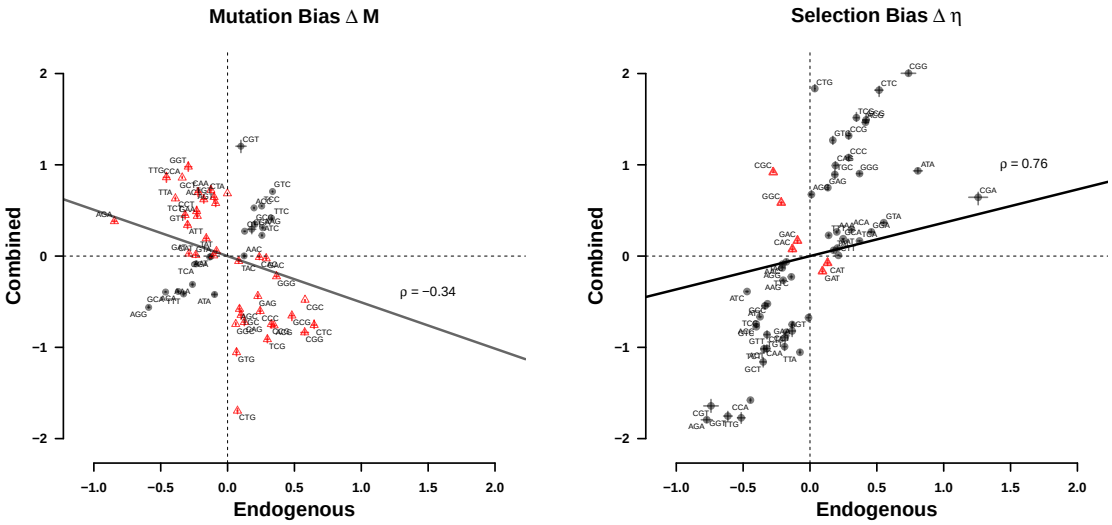


Figure 3.6: Comparison of (a) mutation bias ΔM and (b) selection bias $\Delta\eta$ parameters for endogenous genes and combined gene sets. Estimates are relative to the mean for each codon family. Black dots indicate ΔM or $\Delta\eta$ parameters with the same sign for the endogenous and exogenous genes, red dots indicate parameters with different signs. Black line shows the type II regression line (Sokal and Rohlf, 1981). Dashed lines mark quadrants.

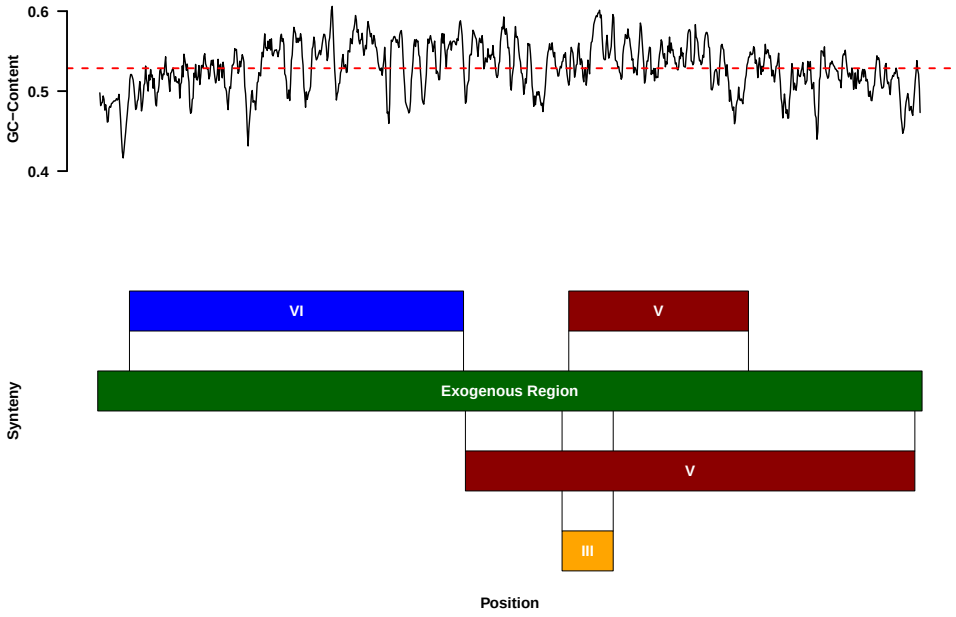


Figure 3.7: Synteny relationship of *E. gossypii* and the exogenous genes. Indicated is the GC content along the introgression.

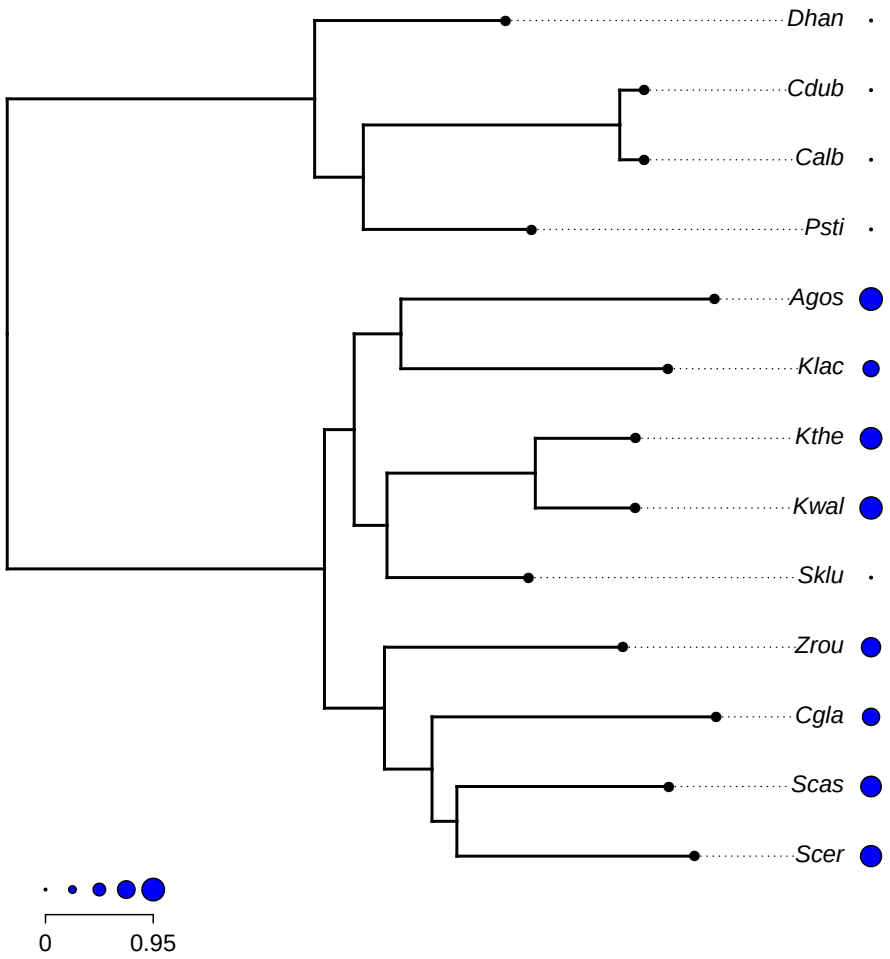


Figure 3.8: Amount of synteny for each species in units of standard deviations for selected species.

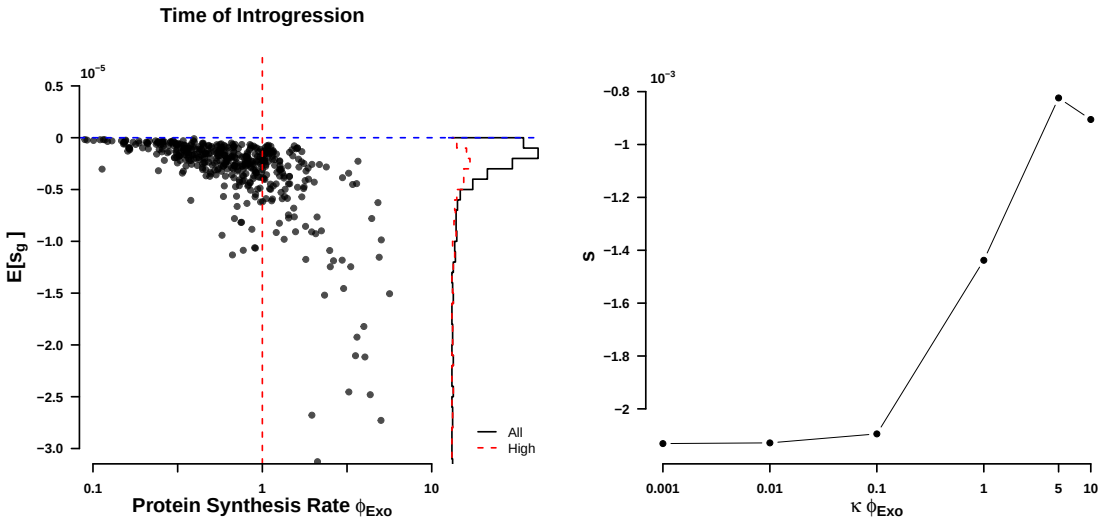


Figure 3.9: Genetic load (left) without scaling of ϕ per gene, and change of total genetic load with scaling κ between *E. gossypii* and *L. kluyveri* (right)

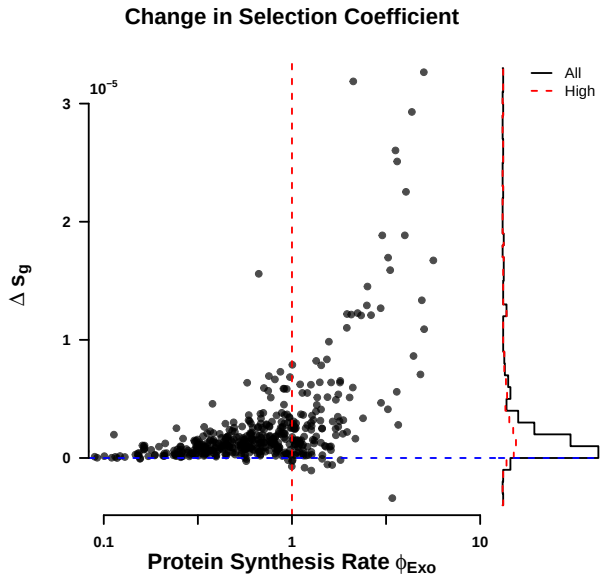


Figure 3.10: Total amount of adaptation estimated to have occurred between time of introgression and currently observed per gene.

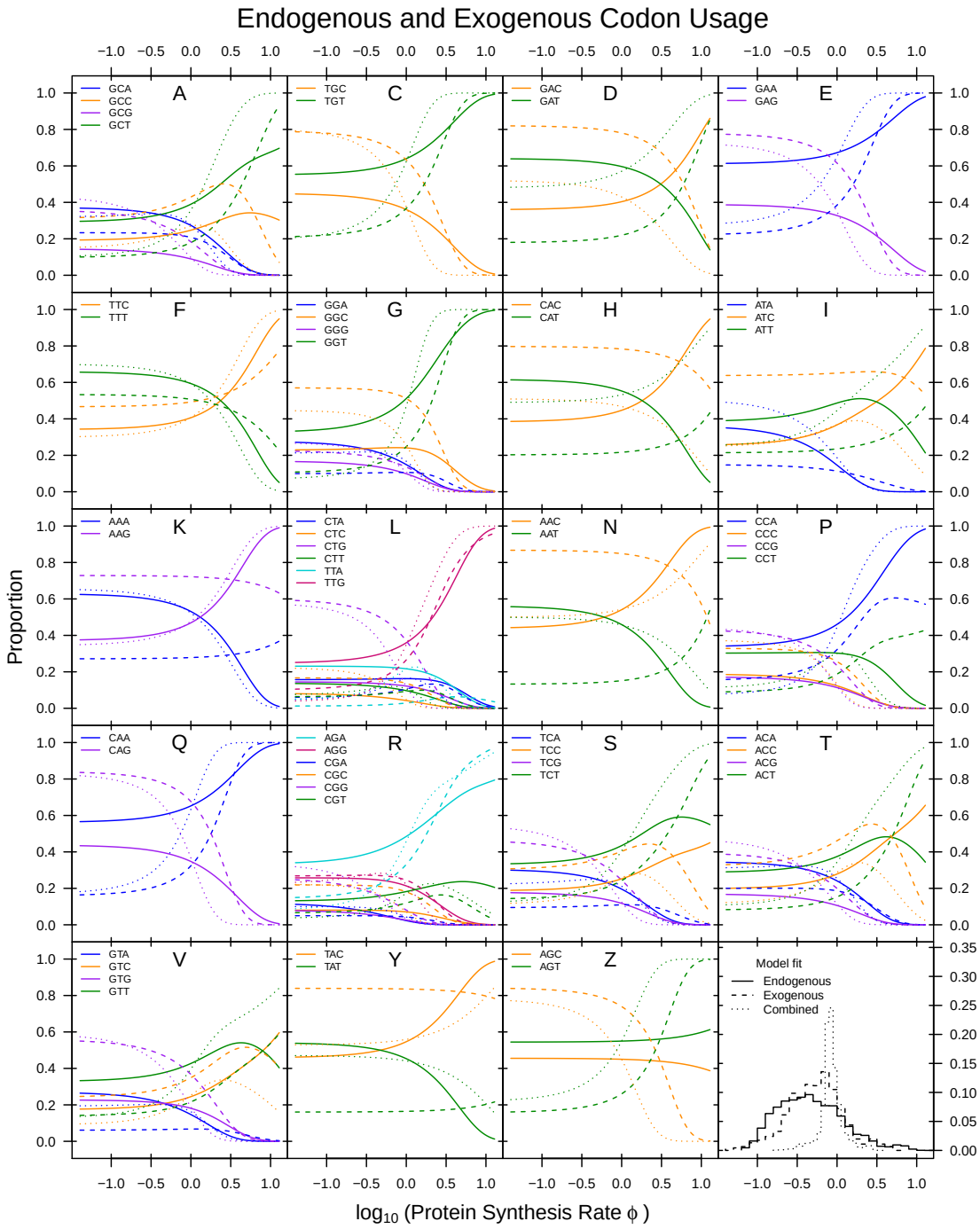


Figure 3.11: Codon usage patterns for 19 amino acids. Amino acids are indicated as one letter code. The amino acids Serine was split into two groups (S and Z) as Serine is coded for by two groups of codons that are separated by more than one mutation. Solid line indicates the endogenous codon usage, dashed line indicates the exogenous codon usage, dotted line indicates the combined codon usage.

Chapter 4

Site specific, physicochemical based phylogenetic models outperform experimentally informed models and overcome their laboratory bias

This chapter is an early version of a paper to be submitted to Genome Biology and Evolution and co-authored with Michael A. Gilchrist and Brian C. O'Meara.

C. Landerer, B.C. O'Meara, M.A. Gilchrist, Phylogenetic model of stabilizing selection is more informative about site specific selection than extrapolation from laboratory estimates

4.1 Abstract

The ever increasing importance of phylogenetics has not been meet with the appropriate development in models. Many commonly used phylogenetic models lack biological realism. Models focused on nucleotides are agnostic to selection on higher level selection on codons or amino acids. Amino acid models on the other hand lack the ability to properly account for mutations. Codon models try to account for both, mutation and selection, but share that only a single substitution matrix is used, resulting in the same equilibrium frequency at each site. Two novel models, *SelAC* and *phydms*, attempt to remedy this issue by either inferring site specific selection from the sequence data or use supplementary information on site specific selection. Here we assess and compare the fit and adequacy of phylogenetic inferences made using supplementary information on selection with *SelAC*, a novel codon model of stabilizing selection on amino acids. We utilize site specific selection parameters for the β-lactamase TEM estimated via deep mutation scanning to supplement phylogenetic inference to 49 observed sequences. Using AIC as a measure of model fit, we find that supplementary selection parameters improve model fit compared to classical models without site specific selection (ΔAIC = 289) but lack model adequacy. We also highlight that this lack in model adequacy is likely due to biased laboratory conditions. In contrast, *SelAC* not only provides improved model fit over classical models without site specific selection (ΔAIC = 871) and experimentally supplemented models (ΔAIC = 582), it also shows

improved model adequacy. This indicates that the development of more realistic models is more promising than the usage of supplementary data for phylogenetic inference.

Phylogenetic inference is of ever increasing importance across biology (O'Meara *et al.*, 2006; Yang and Bourne, 2009; Ruprecht *et al.*, 2017; Schwartz and Schäffer, 2017). Most common models used for phylogenetic inference are incorporated into powerful software packages such as RAxML (Stamatakis, 2014), RevBayes (Höhna *et al.*, 2016), or IQTree (Nguyen *et al.*, 2015) While commonly used models are fast and easy to use, they lack biological realism.

Phylogenetic models focused on the nucleotide composition such as GTR, or UNREST (Tavare, 1986; Yang, 1994) are limited to mutation effects and are agnostic to any higher level selection on codons or amino acids. Amino acid models like JTT (Jones *et al.*, 1992), BLOSSUM (Henikoff and Henikoff, 1992), or WAG (Whelan and Goldman, 2001) attempt to describe the effects of natural selection, however, these do not properly account for mutations between nucleotides and are purely phenomenological. In an attempt to remedy the shortcomings of nucleotide and amino acid models, codon models combine mutation between nucleotides and selection on the amino acids for which they code. Most popular are the codon model by Goldman and Yang (1994) (*GY94*) and its derivatives. However, *GY94* is commonly misinterpreted and provides only a restricted selection scenario that is best described as frequency dependent selection (Hughes and Nei, 1988; Nowak, 2006; Hughes, 2007; Beaulieu *et al.*, 2019).

One common property of the aforementioned models is the fact that they use a single substitution model across all sites. As a result, every site, whether a nucleotide, amino acid, or codon, have the same equilibrium distribution. Biologists, however, have long recognized that equilibrium frequencies and thus the substitution matrix responsible, can vary substantially between sites (Felsenstein, 1981; Gojobori, 1983). Individual sites along the sequence often show differences in evolutionary rates, and wide range of preferences for specific amino acids (Ashenberg *et al.*, 2013; Echave *et al.*, 2016). In response, Halpern and Bruno (1998) (*HB98*) provided a general, codon model were each codon site has its own, distinct substitution matrix. The cost to this generality is the need for 19

amino acid specific selection parameters per site (N.B., the optimal amino acid, by definition, has a selection parameter of 0). This need for estimating a large number of selection parameters from the sequence data makes the application of Halpern and Bruno (1998) model unfeasible for most studies. To overcome this parameterization problem,Bloom (2014) proposed using data from deep mutation scanning experiments (DMS) as a means of estimating the site specific parameters needed in the Halpern and Bruno (1998). Bloom and others (Bloom, 2014, 2017; Hilton *et al.*, 2017) report that using DMS selection parameters greatly improves model fit over models without site specific selection parameters.

The power of DMS stems from the ability to manipulate a large number of individuals in the laboratory and estimate genotype fitness based on frequency changes over many generations. This, however, limits the application of DMS selection parameters for phylogenetic inference to only organisms which can be cultured in the laboratory and with short generation times. More troubling is the fact that variation between DMS experiments can lead to significant differences between model fits (Hilton *et al.*, 2017). In addition, this inter-laboratory variation is likely small compared to the variation between laboratory conditions and those organisms usually encounter in the wild. As a result, *a priori* the value of DMS selection parameters for making inferences about sequences evolution in the wild is questionable.

An alternative to using laboratory based selection parameters to mitigate the parameterization issues introduced in *HB98*, an is to simplify the *HB98* model itself. For example, Lartillot and colleagues mitigate the high numbers of parameters required by *HB98*'s codon model using a site categorization approach where a limited, but *a priori* unspecified, number of site categories are estimated from the sequence data (Lartillot and Philippe, 2004; Le *et al.*, 2008; Rodrigue *et al.*, 2008; Rodrigue and Lartillot, 2014). More recently, (Beaulieu *et al.*, 2019) introduced a new codon model, *SelAC*, where 20 site categories are assumed *a priori* to underlie the *HB98* model. *SelAC* combines physicochemical properties and site specific heterogeneity in the strength of selection together

using a simplistic nested modeling approach. Briefly, *SelAC* infers an optimal amino acid for each site and then estimates the selection parameters for the remaining amino acids based on their physicochemical distance from the optimal amino acid and site specific sensitivity term which is, in turn, treated as a random effect.

We assess and compare the fit and adequacy of phylogenetic inferences made using supplementary DMS selection parameters and *SelAC*. We use *phydms* (Hilton *et al.*, 2017) in order to utilize supplementary DMS selection parameters, fitting 49 TEM sequences observed in natural populations of *E. coli* presented in Bloom (2017). Following Bloom (2017); Hilton *et al.* (2017), we used the DMS based selection parameters from Stiffler *et al.* (2016) for β-lactamase TEM to supplement our phylogenetic inference. TEM is an enzyme found in gram-negative bacteria and catalyzes antibiotics with a β-lactam ring, providing antibiotic resistance (Neu, 1969). The selection pressure imposed during the DMS experiment was limited to ampicillin and focused solely on the variant TEM-1 (Stiffler *et al.*, 2016). However, TEM variants can also confer resistance to a wide range of other antibiotics (Sougakoff *et al.*, 1988, 1989; Goussard *et al.*, 1991; Mabilat *et al.*, 1992; Chanal *et al.*, 1992; Brun *et al.*, 1994).

Using AIC as a measure of model fit, as before we find that *phydms* outperforms the 227 nucleotide and codon models included in the IQTree package (Bloom, 2014, 2017), but that *SelAC* outperforms *phydms* by an additional 582 AIC units. While very large, our estimate of ΔAIC between *SelAC* and *phydms* is likely an under estimate given the fact that we, conservatively, counted each inferred amino acid as a separate parameter when calculating *SelAC*'s AIC value. In addition to a superior fit to the observed data, *SelAC* shows higher model adequacy and implies more realistic values of genetic load than *phydms*. We attribute *phydms*'s poor model adequacy to laboratory bias in the DMS selection parameters. This poor model adequacy, in turn, leads to the unrealistically large estimates of genetic load of the observed TEM sequences.

Together, our results indicate that models can be more informative and applicable than unnatural supplementary data for phylogenetic inference. *SelAC*, in contrast, provides biological meaningful information such as site specific optimal amino acids and estimates of selection parameters. In addition, *SelAC* does not rely on supplementary data, thus making it applicable to any protein coding sequence alignments, and can be expanded to test other hypothesis.

4.2 Results

4.2.1 *SelAC* Outperforms Experimentally Informed Models

We compared *SelAC* and *phydms*, two models of site specific stabilizing selection, to 131 nucleotide and 97 codon uniform site models fitted using IQTree (Nguyen *et al.*, 2015, see Table 4.1 for the best performing models and Table 4.3 for all models). *SelAC* shows the best model fit and provides an improvement of 582 AIC units over the empirically informed model fit by *phydms*. This better performance is in spite of the fact that *SelAC* requires the specification of one optimal amino acid for each of the 263 codons in our TEM alignment. The *phydms* model, parameterized by site specific selection parameters from Stiffler *et al.* (2016) performs second best and provides an improvement of 289 AIC units for the best uniform site model *SYM+R2* (*SYM*) Zharkikh (1994). The best performing uniform codon model is the *GY94+F1X4+R2* variant of *GY94*. In addition to *SYM+R2*, *GY94* is outperformed by 109 other nucleotide models. In contrast to *SelAC*, which is a model of stabilizing selection, *GY94* is best interpreted as frequency dependent selection (Hughes and Nei, 1988; Nowak, 2006; Hughes, 2007; Beaulieu *et al.*, 2019).

It is worth noting that of the 374 parameters estimated by *SelAC*, the vast majority, 263, are discrete parameters corresponding to the optimal amino acid for each site. These 263 discrete parameters are only $\sim 5\%$ of the $19 \times N = 4997$ parameters necessary to fully describe site specific selection. Statistically speaking, it is unclear if *SelAC*'s discrete

Table 4.1: Comparison of the best performing models by category based on their AIC values, where $\log(Lik)$ is the log-likelihood each model and n is the number of model parameters estimated from the aligned sequence data. The two best performing models are the site specific models of amino acid stabilizing selection *SelAC* and *phydms*. The best performing nucleotide model is the variant of Zharkikh (1994)'s symmetrical model *SYM+R2* with two rate categories. The best performing codon model is the *GY94+F1X4+R2* variant with unequal nucleotide frequencies but equal frequencies over all three codon positions and two rate categories. See Table 4.3 for results from all 229 models tested.

Model	$\log(Lik)$	n	AIC	ΔAIC
SelAC	-1498	374	3744	0
phydms	-2061	102	4326	582
SYM+R2	-2206	102	4615	871
GY94+F1X4+R2	-2243	102	4690	946

parameters contribute to the Kullback-Leibler divergence in a similar manner as continuous parameters. As a result, it is possible that we are over penalizing *SelAC* and thus our AIC and ΔAIC values underestimate *SelAC*'s performance.

In addition to variation in model performance, we also observe differences in the topology between model fits. Because the *SelAC* model is too slow in its current form to feasibly identify the best tree topology, we fixed its topology to that estimated using the codon model of Kosiol *et al.* (2007). In order avoid biasing our results towards *SelAC*, we used *SelAC*'s topology as an initial condition of our *phydms* model fitting. The fact that the best fitting *phydms* parameterization diverges from the initial topology indicates our estimate of ΔAIC between *SelAC* and *phydms* is conservative. If we utilize the *phydms* inferred topology with *SelAC*, we find a very similar but slightly worse model fit to our original *SelAC* fit (ΔAIC = 2). This indicates that the topology has less impact on the model fit than the assumed optimal amino acid sequence, likely due to the short branches and many polytomies in the phylogeny (Figure 4.4). Overall, Figure 4.1 shows that the estimated phylogenetic trees shift from long terminal branches (*SelAC*) to longer internal branches (*phydms*). While the *SelAC* model fit shows 84% of all evolution happening along the terminal branches, this reduces 77% in the *phydms* and *GY94* model fits. In addition, polytomies are widespread in all estimated phylogenies, and generally, the shortest branch length are estimated by *SYM*.

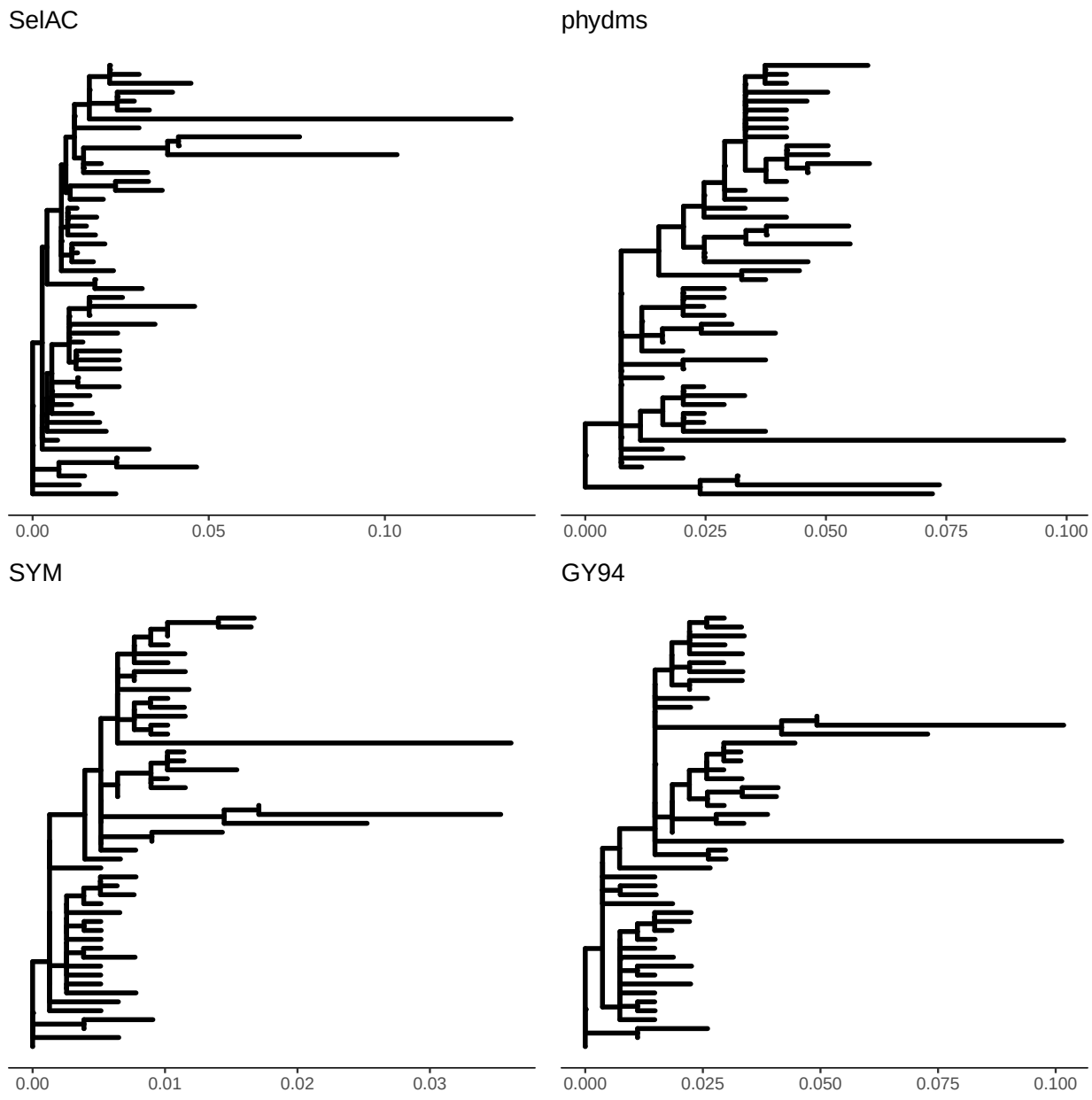


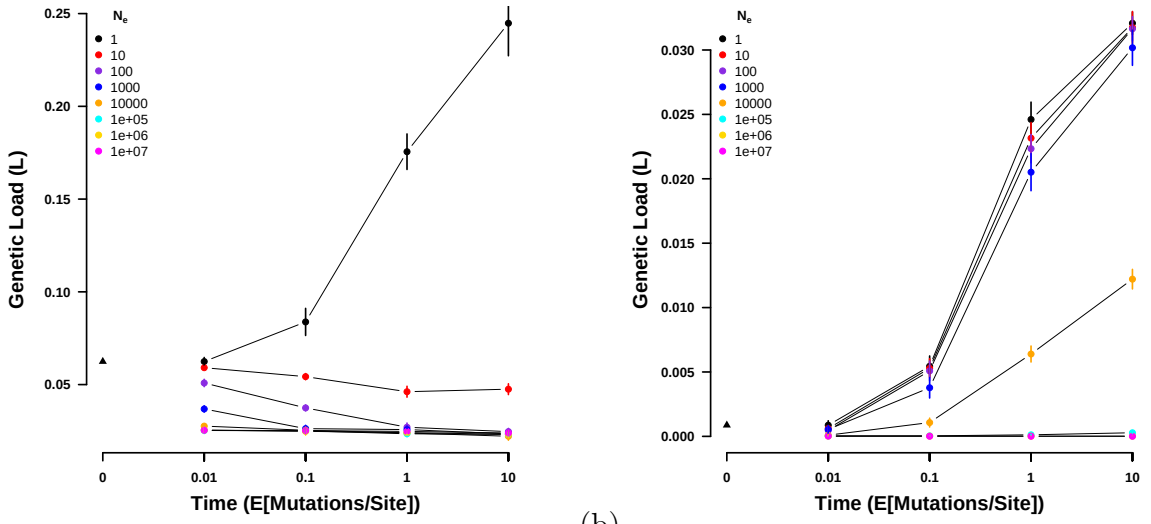
Figure 4.1: Phylogenies resulting from *SelAC*, phydms, *SYM*+R2, and *GY94*+F1X4+R2. As *SelAC* is currently to slow for the inference of topologies, the topology for the *SelAC* phylogeny was inferred using the codon model of Kosiol *et al.* (2007).

4.2.2 DMS is Inconsistent with Genetic Variation in TEM

In order to evaluate the model adequacy of *SelAC* and *phydms*, we calculated the genetic loads of the observed TEM sequences under each model and compared them to their expected values generated by simulating the models. More specifically, we simulated sequences forward in time from the ancestral state under the DMS and *SelAC* inferred selection to establish a point of reference and further assess model adequacy.

Simulations under the *phydms* selection parameters show that the genetic load of the observed sequences is larger than the genetic load of the simulated sequences. The simulation yields an average site specific genetic load of 0.025 (Figure 4.2a). Thus, even under the *phydms* selection parameters, we would expect the observed sequences to show higher adaptation. Simulations under the *SelAC* inferred selection show that the genetic load of the observed sequences is less than the genetic load of the simulated sequences. The simulation yields an average site specific genetic load of 1.3×10^{-5} (Figure 4.2b). This appears to be near the selection-mutation-drift equilibrium and is consistent with theoretical population genetics (see Discussion).

A more detailed analysis shows that we find 100 sites where *SelAC* predicts a genetic load of zero but the DMS estimates predict a non-zero genetic load. All 100 cases show a significant difference in the likelihood between the *SelAC* and the DMS inferred optimal amino acid given the observed sequence data. In addition, these 100 sites show a significant higher average genetic load than the remaining 163 sites of 0.0157 and 0.003, respectively (paired t-test, $p = 3 \times 10^{-13}$). For 52 sites, both *phydms* and *SelAC* estimate a non-zero genetic load. In half the cases, the same optimal amino acid is predicted, in the remaining half *phydms* predicts a significantly different optimal amino acid. Again we find a significant difference in genetic load between the half for which the *SelAC* and *phydms* predictions of the optimal amino acid agree and the half for which they differ of 0.004 and 0.0158, respectively (paired t-test, $p = 2 \times 10^{-5}$). The site specific selection estimated by DMS for the observed TEM sequences represents an average site specific load of 0.065. In contrast, the site specific



(a) (b)

Figure 4.2: Sequences simulated from the ancestral state under the site specific selection on amino acids estimated using DMS (left) and *SelAC* (right) at various times for a range of N_e values. Time units are expected mutations per site, which equals the substitution rate of a neutral mutation. Points indicate sample means and vertical bars indicate standard deviations. Initial sequence is the inferred ancestral state of the TEM variants and indicated by a black triangle.

genetic load estimated by *SelAC* for the observed TEM sequences represents an average site specific genetic load of only 2.4×10^{-7}.

Overall, the *SelAC* fit shows high model adequacy and predicts a zero genetic load at invariant sites. In contrast *phydms* predicts a genetic load at these sites that is not significantly different from the genetic load at the variant sites (Table 4.2). This shows that the distribution of genetic load differs between DMS inferred site specific selection and *SelAC* inferred site specific selection. If we assume the site specific selection parameters used in *phydms*, 111 sites show no genetic load. In contrast, if we assume the site specific selection estimated by *SelAC*, 207 sites show no genetic load. The *SelAC* selection parameters are more in line with the observed sequences as only 66 sites show genetic variation.

The difference in the predictions of genetic load between *SelAC* and DMS are caused by the difference of the *SelAC* and DMS predicted sequences of selectively favored amino acids. While the *SelAC* sequence of selectively favored amino acids has 99% sequence similarity with

Table 4.2: Genetic load at variant and invariant sites in the TEM alignment according to *SelAC* and DMS

		Genetic Load	
Sites	# Residues	*SelAC*	DMS
Variant	66	6.3×10^{-7}	0.010
Invariant	197	0	0.007

the observed consensus sequence, the DMS predicted sequence only shows 52% similarity. We also observe that 46 % of the sites do not show the selectively favored amino acid at all. For example, at site 157, we observe the non-polar, hydrophobic amino acids methionine, but the DMS experiment predicts that the polar, hydrophilic amino acid threonine provides the highest fitness. Together our results suggests that the DMS selection parameters used in *phydms* are not informative about selection in the wild and that *SelAC* is a more appropriate tool to obtain such estimates.

4.3 Discussion

We compared the performance of two codon level phylogenetic models with site specific selection, *phydms* and *SelAC*, as well as 227 more commonly used codon and nucleotide models in explaining 49 aligned TEM sequences obtained from Bloom (2017). Using AIC as measure of model fit we find that both models of site specific selection, *phydms* and *SelAC* perform substantially better that the alternative models (Table 4.1). Further, we find that *SelAC* substantially outperforms *phydms* (ΔAIC = 582).

The improved performance of *phydms* and *SelAC* presumably results from their ability to more realistically describe the effects of natural selection on sequence evolution. However, this realism comes at a cost. *phydms* requires supplementary selection parameters for each amino acid at every site, which necessitates experimental work. *SelAC*, on the other hand, uses a nested modeling approach, which avoids the necessity of amino acid specific selection parameters, but greatly increases the computational cost of model fitting.

We further assess the model adequacy of *phydms* and *SelAC* which we primarily define as the similarity of the sequence of optimal amino acids to the observed consensus sequence. While the consensus sequence ignores the phylogenetic relationship between the sequences, it is a still good metric to assess the realism of the estimated optimal amino acids as it provides a summary of the amino acids observed in the wild. We also assess the genetic load of the observed sequences according to the DMS and *SelAC* selection parameters. Model adequacy is a measure that describes how well observed data can be reproduced by a model and is unfortunately often ignored. Since the model adequacy of *phydms* is a direct function of the supplementary DMS measurements we focus directly on these measurements.

Like model selection, model adequacy strongly favors *SelAC*. When we assess model adequacy by sequence similarity the sequence of optimal amino acids estimated by DMS has only 49 % sequence similarity while the *SelAC* estimated sequence shows a sequence similarity of 99 % with the observed consensus sequence. Given these results, it is tempting to assume that the consensus sequence will always fair best, however, this would assume independence between the observed sequences. Furthermore, the high sequence similarity between the consensus sequence and the sequence of optimal amino acids is likely due to the high average sequence similarity in the TEM alignment of 98 %.

Similarly, we find the genetic load of the DMS sequences is substantially higher when assuming the DMS estimated selection on amino acids compared to the *SelAC* estimates with 0.065 and 2.4×10^{-7}, respectively. However, if we assume that the DMS inference adequately reflects the evolution of TEM in the wild the observed sequences are either maladapted or were unable to reach a fitness peak. This, however is unlikely as *E. coli* has a large effective population size. Estimates are on the order of 10^8 to 10^9 (Ochman and Wilson, 1987; Hartl *et al.*, 1994). We would therefore expect that *E. coli* can effectively explore the sequence space. More specifically, assuming a mutation rate of 2.54×10^{-10} mutations per generation per nucleotide (Lee *et al.*, 2012), we would expect to find 2×10^{-7} mutations per generation in the 789 nucleotide long TEM sequence. This results in between $\mu N_e = 10^1$ to

10^2 mutations per generations throughout the population. The average site specific selection against the observed TEM sequences is $s = 0.085$, we would therefore expect that such a mutation should fix on average within $(4/|s|) \times \ln(2N_e) \sim 1200$ to 1300 generations (Crow and Kimura, 1970). Given *E. coli*'s rapid doubling time of 15 hours in the wild (Gibson *et al.*, 2018), this means that such a mutation should spread through the population in 1.5 years. However, this also clearly shows that the weak mutation assumptions made by all considered models, including *SelAC*, is clearly violated.

In contrast, estimates of selection obtained by *SelAC* show the observed sequences near a fitness peak. This is consistent with predictions discussed above. It, therefore, appears that DMS reflects the biased laboratory selection on the TEM sequences with respect to only one antibiotic, ampicillin. This may be appropriate to model selection in a hospital environment, but not the evolution of TEM in the wild. The evidence we derive from population genetics theory has us expecting the observed sequences at the selection-mutation-drift equilibrium. This, however, is clearly not the case if we assume the DMS inferred selection.

Besides relatively poor performance in terms of model adequacy, DMS has additional shortcomings that limit its use in phylogenetic studies. Like with any other experiment, results can greatly vary between laboratories. Hilton *et al.* (2017) showed that a similar experiment to the one used here by Firnberg *et al.* (2014) performed worse in explaining the observed TEM data. DMS experiments are also costly and limited to microorganisms that can be cultivated and manipulated under laboratory conditions. These laboratory conditions can lead to the bias in selection parameters we show in this study.

The artificial selection environment in the laboratory leads to a very heterogeneous population and very large selection coefficients s unlikely to be observed in the wild. The very large single selection pressure may be the easiest issue to overcome in DMS experiments as it may be possible to include a multitude of weaker selective forces. However, this is often not the goal when performing DMS experiments as they were designed to identify mutational effects in responds to specific selection. *SelAC* on the other hand, better explains the

evolution of TEM in the wild and does not require selection parameters but instead provides such estimates from the sequence data. This makes *SelAC* also applicable to any set of aligned protein coding sequences.

However, *SelAC* is not without shortcomings itself, but its mechanistic nature provides direct avenues to overcome many of them via model expansion. For example, *SelAC* assumes invariance in the optimal amino acid at a site across the whole phylogeny. This may be appropriate for closely related organisms in similar environments but not necessarily for distantly related species. Incorporation of a hidden markov model would not only allow for shifts in the optimal amino acid along the phylogeny. A hidden markov approach would also allow for frequency dependent selection like *GY94*. Frequency dependent selection may be appropriate for certain TEM sites as it plays a role in chemical conflicts between microorganisms. However, our results and the high number of invariant sites in the alignment indicate that such frequency dependent selection may only apply to a small number of sites.

In order to map the amino acid sequence to protein fitness *SelAC* uses the euclidean distance in physicochemical space between amino acids. A more realistic mapping could be employed by adding higher order terms or by utilizing an explicit molecular model. *SelAC* also currently ignores selection on synonymous codon usage and therefore treats synonymous mutations as neutral.

Other shortcomings of *SelAC* with a less clear solution include the relaxation of the constrained that the site specific sensitivity of selection has to be positive. Allowing for a negative sensitivity term would extend *SelAC* to diversifying selection on amino acids. The inclusion of a mixture model where model parameters vary between site categories would allow to distinguish e.g. sites under stabilizing and diversifying selection. Finally, *SelAC*, like all other models considered here, assumes site independence, and thus ignores epistatic interaction between amino acids.

DMS experiments have been proposed to supplement information on selection on amino acids in phylogenetic studies. Our study shows that this information on site specific selection

parameters is unnecessary. This is because the relevant information on stabilizing selection is already embedded within protein coding sequence alignments and can be inferred using a nested modeling approach. In addition to being unnecessary, we show that DMS estimates of selection parameters are unnaturally biased towards laboratory conditions. The ability to expand *SelAC* as outlined above make it a valid starting point for such improvements and allow for explicit hypothesis testing. Taken together, our results indicate that efforts in improving phylogenetic inferences are likely better spent on the development of more realistic models rather than generating and/or incorporating DMS data.

4.4 Materials and Methods

4.4.1 Phylogenetic Inference and Model selection

Aligned TEM sequences were obtained from Bloom (2017) Experimentally selection parameters for TEM were taken from Stiffler *et al.* (2016). We followed (Bloom, 2017) and used the experimental selection parameters to determine site specific equilibrium frequencies for *phydms*. *phydms* (version 2.5.1) was fitted to the site specific selection from Stiffler *et al.* (2016) using python (version 3.6). *SelAC* (version 1.6.1) was fitted to the TEM alignment using R (version 3.4.1) (R Core Team, 2015). We assumed the physicochemical properties estimated by Grantham (1974) and only estimated the weighting terms for each property from the data. We choose the constraint free, general unrestricted model (UNREST) (Yang, 1994) as mutation model for *SelAC*. All other models were fitted using IQTree (Nguyen *et al.*, 2015). All models were fitted using maximum likelihood. We report each model's log likelohood ($\log(Lik)$), and AIC.

4.4.2 Sequence Simulation

Sequences were simulated by stochastic simulations using a Gillespie algorithm (Gillespie, 1976). Fixation probabilities were based on Sella and Hirsh (2005). The selection

parameters were estimated using *SelAC* or taken from Stiffler *et al.* (2016). We choose the selection parameters resulting from the highest concentration (2500 $\mu g/mL$) treatment of ampicillin for our comparison. We rescaled the experimental selection parameters such that the amino acid with the highest fitness at each site has a value of one. Mutation rates for the simulations were taken from the *SelAC*. The initial sequences was the ancestral sequence reconstructed using FastML (Ashkenazy *et al.*, 2012) (last accessed: 30.09.2018). Each sequence was simulated 10 times and we report average genetic load and sequence similarity and the standard error. The sequences were sampled at times 0.01, 0.1, 1, and 10 expected mutations per site.

4.4.3 Estimating site specific selection parameters w_i

Following Beaulieu *et al.* (2019) w_i is proportional to

$$w_i \propto \exp(-A_0 \eta \psi) \tag{4.1}$$

where A_0 describes the decline in fitness with each high energy phosphate bond wasted per unit time, and ψ is the protein's production rate. η is the cost/benefit ratio of a protein (see (Beaulieu *et al.*, 2019) for details). However, *SelAC* only estimates a composition parameter $\psi' = A_0 \psi N_e$ thus

$$\psi = \frac{\psi'}{A_0 N_e q} \tag{4.2}$$

SelAC assumes that the effective population size $N_e = 5 \times 10^6$ and that $A_0 = 4 \times 10^{-7}$ (Gilchrist, 2007).

4.4.4 Model Adequacy

Model adequacy was assessed by comparing the observed sequences and simulations under the site specific selection inferred by the deep mutation scanning experiment or *SelAC*. First, similarity between the sequence of selectively favored amino acids and the observed TEM

sequences was assessed. Sequence similarity was measured as the number of differences in the aligned amino acid sequences. Second, the genetic load of the observed and the simulated sequences was calculated using either the site specific selection inferred by the deep mutation scanning experiment or *SelAC*. The average genetic load for site i in the alignment was calculated as

$$L_i = \frac{w_{max,i} - \overline{w_i}}{w_{max,i}} \tag{4.3}$$

where $w_{max,i}$ is the fitness of the selectively favored amino acids at position i, either estimated using the site specific selection inferred by DMS or *SelAC*. $\overline{w_i}$ represents the average fitness of the residues observed at position i. The average sequence specific genetic load L was calculated as the sum of the site specific genetic loads $L = \frac{1}{n}\sum_{i=1}^{n} L_i$ where n is the number of amino acid sites.

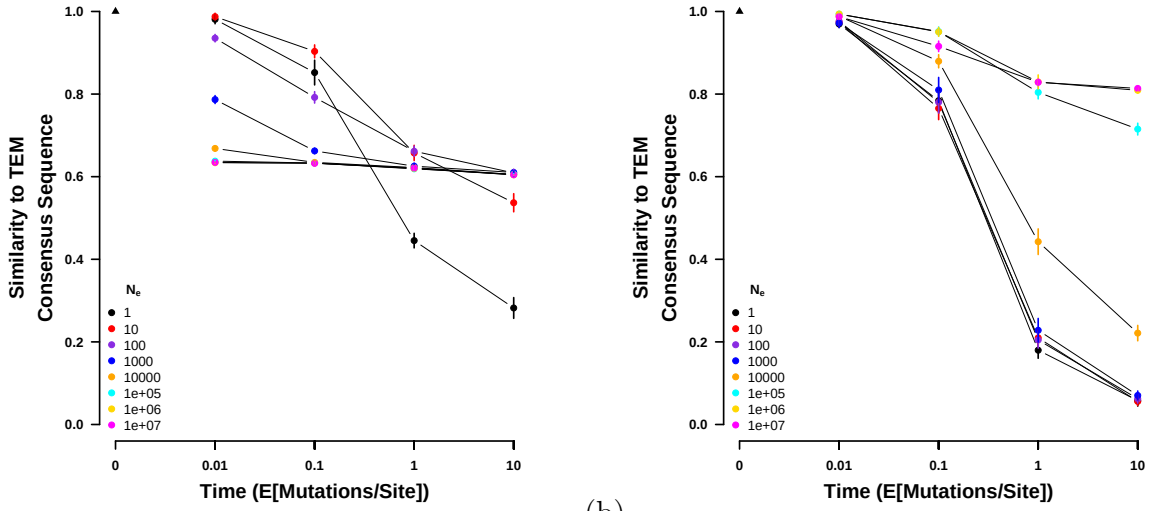
4.5 Acknowledgments

This work was supported in part by NSF Award and DEB-1355033 (BCO, MAG, and Russell Zaretzki) with additional support from the University of Tennessee Knoxville. CL received additional support as a Graduate Student Fellow at the National Institute for Mathematical and Biological Synthesis, an Institute sponsored by the National Science Foundation through NSF Award DBI-1300426, with additional support from UTK. The authors would like to thank Russell Zaretzki, Jeremy Beaulieu, and Alexander Cope for their helpful criticisms and suggestions for this work.

4.6 Appendix: Supplementary Material

The *SelAC* inferred sequence of selectively favored amino acids has 99% sequence similarity with the observed consensus sequence. This may not be to surprising given that *SelAC* only uses the sequence data and no experimental supplementary data. Simulations support the better model adequacy of *SelAC*, however the decline in sequences sequence similarity to 83%, indicating that *SelAC* underestimates the strength of selection (Figure 4.3b).

The sequence of selectively favored amino acids experimentally estimated by DMS shows a low sequence similarity of only 52% with the observed consensus sequence. We find that the selectively favored amino acid estimated by DMS is not found in the wild in 46.4 % of sites. Additionally, the physicochemical properties appear to differ between the observed and the DMS estimated optimal amino acids. Simulations of codon sequences under the DMS inferred site specific selection further highlights that we would not expect to see the observed TEM sequences under these conditions. We find that the simulated sequences show up to 62% sequence similarity to the observed consensus sequence (Figure 4.3a). This is a 10 % higher sequence similarity than the sequence of selectively favored amino acids estimated by DMS have with the observed consensus sequence.



(a) (b)

Figure 4.3: Sequences simulated from the ancestral state under the site specific selection on amino acids estimated using DMS (left) and *SelAC* (right) at various times for a range of N_e values. Time units are expected mutations per site, which equals the substitution rate of a neutral mutation. Points indicate sample means and vertical bars indicate standard deviations. Initial sequence is the inferred ancestral state of the TEM variants and indicated by a black triangle.

Table 4.3: Model selection of 229 models of nucleotide and codon evolution.

No.	Model	log(*Lik*)	n	AIC	ΔAIC
1	*SelAC* +G4	-1497.971	374	3743.942	0
2	*phydms*	-2060.85	102	4325.7	582
3	SYM+R2	-2205.877	102	4615.754	871.754
4	TIMe+R2	-2232.406	100	4664.811	920.811
5	TVMe+R2	-2232.838	101	4667.677	923.677
6	TIM3e+R2	-2234.332	100	4668.664	924.664
7	TIM2e+R2	-2234.381	100	4668.763	924.763
8	K3P+R2	-2235.777	99	4669.553	925.553
9	TNe+R2	-2236.078	99	4670.155	926.155
10	SYM+R3	-2229.616	104	4667.232	923.232
11	TIM+F+R2	-2230.958	103	4667.915	923.915
12	TIMe+R3	-2232.404	102	4668.808	924.808
13	GTR+F+R2	-2228.537	105	4667.073	923.073
14	K3Pu+F+R2	-2232.617	102	4669.234	925.234
15	TVM+F+R2	-2230.105	104	4668.21	924.21
16	TVMe+R3	-2232.838	103	4671.676	927.676
17	K2P+R2	-2239.424	98	4674.847	930.847
18	TIM3e+R3	-2234.332	102	4672.664	928.664
19	TIM2e+R3	-2234.381	102	4672.762	928.762
20	TIM3+F+R2	-2233.064	103	4672.127	928.127
21	TIM2+F+R2	-2233.114	103	4672.227	928.227
22	K3P+R3	-2235.777	101	4673.553	929.553
23	TN+F+R2	-2234.624	102	4673.249	929.249
24	TPM3u+F+R2	-2234.673	102	4673.347	929.347
25	TPM3+F+R2	-2234.674	102	4673.348	929.348
26	TPM2u+F+R2	-2234.681	102	4673.363	929.363
27	TPM2+F+R2	-2234.683	102	4673.365	929.365
28	TNe+R3	-2236.077	101	4674.155	930.155
29	TIM+F+R3	-2230.958	105	4671.915	927.915
30	HKY+F+R2	-2236.266	101	4674.531	930.531
31	GTR+F+R3	-2228.536	107	4671.073	927.073
32	K3Pu+F+R3	-2232.617	104	4673.234	929.234
33	TVM+F+R3	-2230.105	106	4672.21	928.21
34	K2P+R3	-2239.192	100	4678.384	934.384
35	TIM3+F+R3	-2233.063	105	4676.127	932.127
36	TIM2+F+R3	-2233.113	105	4676.227	932.227
37	TN+F+R3	-2234.624	104	4677.249	933.249
38	TPM3u+F+R3	-2234.673	104	4677.347	933.347
39	TPM3+F+R3	-2234.674	104	4677.348	933.348
40	TPM2u+F+R3	-2234.681	104	4677.363	933.363

Table 4.3 Continued

No.	Model	log(Lik)	n	AIC	ΔAIC
41	TPM2+F+R3	-2234.682	104	4677.364	933.364
42	HKY+F+R3	-2236.074	103	4678.148	934.148
43	SYM+I+G4	-2243.212	102	4690.424	946.424
44	TVMe+I+G4	-2244.533	101	4691.066	947.066
45	TIMe+I+G4	-2246.457	100	4692.914	948.914
46	K3P+I+G4	-2248.166	99	4694.332	950.332
47	TVM+F+I+G4	-2241.853	104	4691.707	947.707
48	TIM3e+I+G4	-2247.379	100	4694.758	950.758
49	K3Pu+F+I+G4	-2245.156	102	4694.311	950.311
50	GTR+F+I+G4	-2241.484	105	4692.968	948.968
51	TIM+F+I+G4	-2244.418	103	4694.836	950.836
52	TPM3u+F+I+G4	-2246.03	102	4696.06	952.06
53	TPM3+F+I+G4	-2246.069	102	4696.138	952.138
54	TIM2e+I+G4	-2248.934	100	4697.868	953.868
55	TNe+I+G4	-2250.587	99	4699.174	955.174
56	TIM3+F+I+G4	-2245.534	103	4697.068	953.068
57	K2P+I+G4	-2252.181	98	4700.362	956.362
58	TPM2u+F+I+G4	-2247.579	102	4699.158	955.158
59	TPM2+F+I+G4	-2247.685	102	4699.371	955.371
60	HKY+F+I+G4	-2249.065	101	4700.13	956.13
61	TIM2+F+I+G4	-2247.009	103	4700.018	956.018
62	TN+F+I+G4	-2248.511	102	4701.023	957.023
63	TVMe+I	-2254.804	100	4709.608	965.608
64	K3P+I	-2257.72	98	4711.439	967.439
65	SYM+I	-2254.11	101	4710.221	966.220
66	TIMe+I	-2257.074	99	4712.149	968.149
67	TVM+F+I	-2252.157	103	4710.315	966.315
68	K3Pu+F+I	-2254.856	101	4711.712	967.712
69	TIM3e+I	-2257.796	99	4713.592	969.592
70	TPM3+F+I	-2255.771	101	4713.543	969.543
71	TPM3u+F+I	-2255.771	101	4713.543	969.543
72	K2P+I	-2261.218	97	4716.436	972.436
73	GTR+F+I	-2252.067	104	4712.133	968.133
74	TIM+F+I	-2254.783	102	4713.566	969.566
75	TNe+I	-2260.579	98	4717.158	973.158
76	TIM3+F+I	-2255.684	102	4715.368	971.368
77	HKY+F+I	-2258.352	100	4716.703	972.703
78	TIM2e+I	-2259.878	99	4717.757	973.757
79	TVMe+G4	-2258.853	100	4717.705	973.705
80	SYM+G4	-2257.573	101	4717.146	973.146
81	TPM2+F+I	-2257.712	101	4717.423	973.423
82	TPM2u+F+I	-2257.712	101	4717.423	973.423

Table 4.3 Continued

No.	Model	log(*Lik*)	n	AIC	ΔAIC
83	K3P+G4	-2261.922	98	4719.844	975.844
84	TIMe+G4	-2260.683	99	4719.365	975.365
85	TN+F+I	-2258.28	101	4718.561	974.561
86	TIM3e+G4	-2261.255	99	4720.51	976.51
87	TVM+F+G4	-2256.108	103	4718.216	974.216
88	TIM2+F+I	-2257.643	102	4719.286	975.286
89	K3Pu+F+G4	-2258.971	101	4719.941	975.941
90	TPM3u+F+G4	-2259.716	101	4721.433	977.433
91	TPM3+F+G4	-2259.717	101	4721.434	977.434
92	GTR+F+G4	-2255.75	104	4719.5	975.5
93	TIM+F+G4	-2258.638	102	4721.276	977.276
94	K2P+G4	-2265.454	97	4724.907	980.907
95	TNe+G4	-2264.219	98	4724.437	980.437
96	TIM3+F+G4	-2259.366	102	4722.732	978.732
97	TIM2e+G4	-2263.57	99	4725.141	981.141
98	JC+R2	-2266.233	97	4726.466	982.466
99	F81+F+R2	-2262.327	100	4724.654	980.654
100	HKY+F+G4	-2262.499	100	4724.999	980.999
101	TPM2+F+G4	-2261.915	101	4725.829	981.829
102	TPM2u+F+G4	-2261.915	101	4725.829	981.829
103	TN+F+G4	-2262.169	101	4726.338	982.338
104	TIM2+F+G4	-2261.585	102	4727.17	983.17
105	F81+F+R3	-2262.028	102	4728.056	984.056
106	JC+R3	-2265.997	99	4729.994	985.994
107	F81+F+I+G4	-2274.845	100	4749.69	1005.69
108	JC+I+G4	-2279.318	97	4752.636	1008.636
109	F81+F+I	-2283.56	99	4765.119	1021.119
110	JC+I	-2287.984	96	4767.968	1023.968
111	F81+F+G4	-2287.834	99	4773.669	1029.669
112	JC+G4	-2292.095	96	4776.19	1032.19
113	*GY94* +F1X4+R2	-2242.963	102	4689.926	945.926
114	MGK+F1X4+R2	-2243.111	102	4690.221	946.221
115	*GY94* +F1X4+R3	-2238.022	104	4684.043	940.043
116	MGK+F3X4+R2	-2229.923	108	4675.846	931.846
117	*GY94* +F1X4+I+G4	-2247.179	102	4698.359	954.359
118	MGK+F1X4+I+G4	-2247.292	102	4698.583	954.583
119	MGK+F1X4+R3	-2241.989	104	4691.978	947.978
120	MGK+F3X4+R3	-2224.78	110	4669.559	925.559
121	*GY94* +F1X4+G4	-2251.144	101	4704.287	960.287
122	MGK+F1X4+G4	-2251.472	101	4704.944	960.944
123	*GY94* +F3X4+R3	-2227.048	110	4674.096	930.096
124	*GY94* +F3X4+R2	-2233.068	108	4682.136	938.136

Table 4.3 Continued

No.	Model	log(*Lik*)	n	AIC	ΔAIC
125	MGK+F3X4+I+G4	-2233.539	108	4683.078	939.0781
126	MGK+F3X4+G4	-2237.512	107	4689.024	945.024
127	*GY94* +F3X4+I+G4	-2238.243	108	4692.485	948.485
128	*GY94* +F3X4+R4	-2227.106	112	4678.213	934.213
129	*GY94* +F3X4+G4	-2242.394	107	4698.789	954.789
130	*GY94* +F1X4+I	-2260.085	101	4722.169	978.169
131	MGK+F1X4+I	-2260.345	101	4722.69	978.69
132	MGK+F3X4+I	-2246.112	107	4706.225	962.225
133	MG+F1X4+R2	-2268.482	101	4738.963	994.963
134	*GY94* +F3X4+I	-2252.532	107	4719.064	975.064
135	MG+F3X4+R2	-2254.453	107	4722.906	978.906
136	MG+F1X4+I+G4	-2272.057	101	4746.113	1002.113
137	MG+F1X4+R3	-2267.523	103	4741.047	997.047
138	MG+F1X4+G4	-2276.171	100	4752.342	1008.342
139	MG+F3X4+I+G4	-2257.945	107	4729.891	985.891
140	MG+F3X4+G4	-2261.949	106	4735.898	991.898
141	MG+F3X4+R3	-2253.514	109	4725.027	981.027
142	SYM	-2329.878	100	4859.756	1115.756
143	TIMe	-2333.105	98	4862.21	1118.21
144	TIM3e	-2333.481	98	4862.961	1118.961
145	TVMe	-2333.164	99	4864.328	1120.328
146	GTR+F	-2328.404	103	4862.809	1118.809
147	K3P	-2336.391	97	4866.783	1122.783
148	MG+F1X4+I	-2284.946	100	4769.892	1025.892
149	TVM+F	-2330.086	102	4864.172	1120.172
150	TIM+F	-2331.48	101	4864.96	1120.96
151	TNe	-2336.729	97	4867.458	1123.458
152	K3Pu+F	-2333.162	100	4866.323	1122.323
153	TIM3+F	-2331.971	101	4865.942	1121.942
154	TPM3+F	-2333.648	100	4867.297	1123.297
155	TPM3u+F	-2333.648	100	4867.297	1123.297
156	TIM2e	-2336.292	98	4868.584	1124.584
157	MG+F3X4+I	-2270.442	106	4752.885	1008.885
158	K2P	-2340.015	96	4872.03	1128.03
159	TN+F	-2335.102	100	4870.204	1126.204
160	HKY+F	-2336.783	99	4871.566	1127.566
161	TIM2+F	-2334.7	101	4871.401	1127.401
162	TPM2u+F	-2336.381	100	4872.761	1128.761
163	TPM2+F	-2336.381	100	4872.762	1128.762
164	JC	-2366.286	95	4922.571	1178.571
165	F81+F	-2362.554	98	4921.108	1177.108
166	*GY94* +F1X4	-2315.788	100	4831.575	1087.575

Table 4.3 Continued

No.	Model	log(*Lik*)	n	AIC	ΔAIC
167	KOSI07+FU+R2	-2325.725	97	4845.45	1101.45
168	MGK+F1X4	-2318.048	100	4836.095	1092.095
169	KOSI07+FU+R3	-2323.063	99	4844.126	1100.126
170	MGK+F3X4	-2304.357	106	4820.713	1076.713
171	*GY94* +F3X4	-2306.17	106	4824.339	1080.339
172	KOSI07+FU+I+G4	-2335.554	97	4865.108	1121.108
173	KOSI07+FU+G4	-2339.513	96	4871.026	1127.026
174	KOSI07+F3X4+R2	-2315.814	106	4843.627	1099.627
175	KOSI07+F3X4+R3	-2310.509	108	4837.018	1093.018
176	KOSI07+F1X4+R2	-2333.491	100	4866.983	1122.983
177	KOSI07+F1X4+R3	-2328.692	102	4861.383	1117.383
178	SCHN05+FU+R2	-2344.705	97	4883.411	1139.411
179	KOSI07+F1X4+I+G4	-2337.965	100	4875.93	1131.93
180	KOSI07+F1X4+G4	-2341.156	99	4880.312	1136.312
181	SCHN05+FU+R3	-2341.179	99	4880.358	1136.358
182	KOSI07+FU+I	-2349.617	96	4891.233	1147.233
183	KOSI07+F3X4+I+G4	-2323.767	106	4859.534	1115.534
184	MG+F1X4	-2342.797	99	4883.593	1139.593
185	KOSI07+F3X4+G4	-2327.376	105	4864.751	1120.751
186	MG+F3X4	-2328.539	105	4867.078	1123.078
187	SCHN05+F1X4+R3	-2340.927	102	4885.854	1141.854
188	KOSI07+F1X4+I	-2349.1	99	4896.2	1152.2
189	SCHN05+F3X4+R3	-2324.472	108	4864.944	1120.944
190	SCHN05+FU+I+G4	-2354.523	97	4903.046	1159.046
191	SCHN05+F1X4+R2	-2348.226	100	4896.452	1152.452
192	SCHN05+F3X4+R2	-2331.916	106	4875.833	1131.833
193	SCHN05+FU+G4	-2358.682	96	4909.365	1165.365
194	KOSI07+F3X4+I	-2336.826	105	4883.653	1139.653
195	SCHN05+F1X4+I+G4	-2351.096	100	4902.192	1158.192
196	SCHN05+F1X4+G4	-2353.895	99	4905.79	1161.79
197	SCHN05+F1X4+R4	-2340.593	104	4889.187	1145.187
198	SCHN05+F3X4+R4	-2324.102	110	4868.203	1124.203
299	SCHN05+F3X4+I+G4	-2338.345	106	4888.69	1144.69
200	SCHN05+F3X4+G4	-2341.811	105	4893.621	1149.621
201	SCHN05+FU+I	-2370.471	96	4932.943	1188.943
202	SCHN05+F1X4+I	-2363.696	99	4925.391	1181.391
203	SCHN05+F3X4+I	-2352.81	105	4915.621	1171.621
204	KOSI07+FU	-2394.782	95	4979.563	1235.563
205	KOSI07+F1X4	-2398.44	98	4992.88	1248.88
206	KOSI07+F3X4	-2383.159	104	4974.318	1230.318
207	SCHN05+FU	-2419.333	95	5028.665	1284.665
208	SCHN05+F1X4	-2416.544	98	5029.088	1285.088

Table 4.3 Continued

No.	Model	log(*Lik*)	n	AIC	ΔAIC
209	SCHN05+F3X4	-2402.838	104	5013.675	1269.675
210	*GY94* +F+R2	-2208.59	159	4735.181	991.181
211	*GY94* +F+G4	-2217.694	158	4751.388	1007.388
212	*GY94* +F+I+G4	-2213.659	159	4745.319	1001.319
213	*GY94* +F+R3	-2202.599	161	4727.198	983.198
214	*GY94* +F+I	-2228.346	158	4772.691	1028.691
215	*GY94* +F+R4	-2202.61	163	4731.219	987.219
216	*GY94* +F	-2282.254	157	4878.509	1134.509
217	KOSI07+F+R2	-2291.643	157	4897.286	1153.286
218	KOSI07+F+G4	-2301.662	156	4915.325	1171.325
219	KOSI07+F+I+G4	-2298.418	157	4910.835	1166.835
220	KOSI07+F+R3	-2286.723	159	4891.446	1147.446
221	KOSI07+F+I	-2311.78	156	4935.559	1191.559
222	SCHN05+F+R2	-2310.015	157	4934.03	1190.03
223	SCHN05+F+G4	-2316.684	156	4945.369	1201.369
224	SCHN05+F+I+G4	-2313.733	157	4941.467	1197.467
225	SCHN05+F+R3	-2303.732	159	4925.463	1181.463
226	SCHN05+F+I	-2327.127	156	4966.254	1222.254
227	SCHN05+F+R4	-2303.45	161	4928.9	1184.9
228	KOSI07+F	-2357.579	155	5025.157	1281.157
229	SCHN05+F	-2379.264	155	5068.528	1324.528

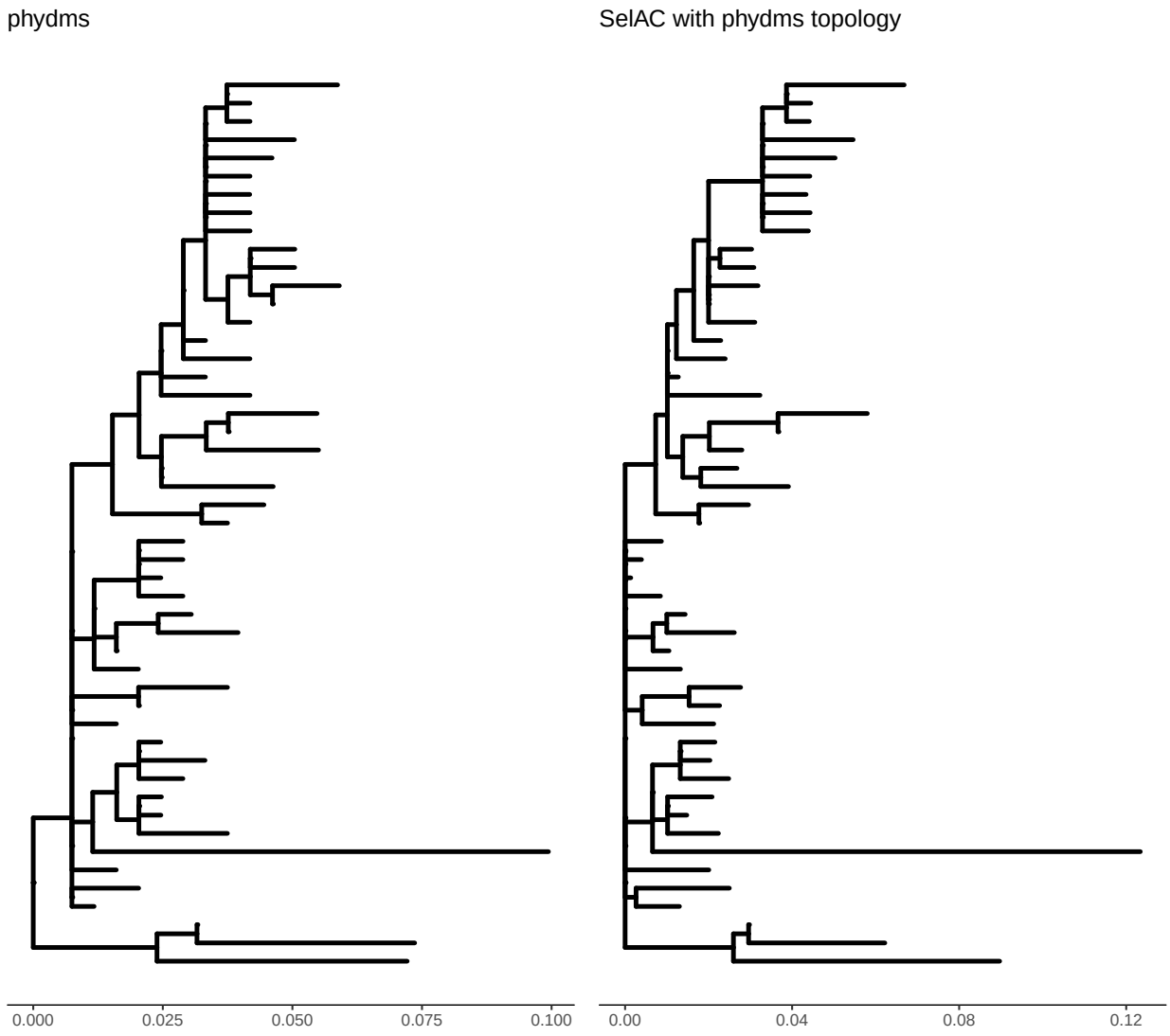


Figure 4.4: Phylogenies resulting from phydms, and *SelAC* using the *phydms* topology.

Chapter 5

Conclusion

5.1 Summary

Protein synthesis from mRNA is the metabolically most expensive process a cell performs with about 20% of the cells total energy budget (Reeds *et al.*, 1985; Waterlow and Millward, 1989). The direct cost for the translation of a protein of length L requires $4L+4$ high energy phosphate bonds provided by ATP and GTP molecules. Protein synthesis is the results of a complex interplay of many different metabolic and regulatory pathways. Each step of protein synthesis is under selection and prone to errors with consequences for downstream processes. This enormous energy expenditure for the translation of a protein from mRNA leads to strong selection for efficient translation (Gilchrist, 2007; Drummond and Wilke, 2008; Gilchrist *et al.*, 2009; Shah and Gilchrist, 2011a; Gilchrist *et al.*, 2015). However, the efficacy of selection varies with the effective population size N_e between organisms, the rate of protein synthesis, and absolute difference in metabolic expenditure with changes in amino acid and codon usage.

On the other hand, proteins are involved in almost all processes a cell performs. From communication between cells, over the processing of metabolites, to the transport of nutrients. This ratio of cost to benefit is the fundamental concept I applied to understand and separate the effects mutation, selection, and genetic drift have on protein sequence evolution. I approached cost and benefit by applying mechanistic models rooted in first

principles to protein coding sequences. In chapter 3, I focused on the cost of protein synthesis and explored the effects of mismatched codon usage. In chapter 4, I focused on the benefit of protein synthesis and estimated site specific selection on amino acids and assessed their adequacy.

5.1.1 The Value of Mechanistic Models

Mathematical and statistical models exist on a spectrum from descriptive over phenomenological to mechanistic with increasing power to extract information from data. Models allow us to summarize data and identify patterns. They are an essential tool to formalize verbal theory and allow for hypothesis testing. Well formulated models grounded in first principles can provide insights into underlying biological processes. Yet, we still have blackboxes in our models and many phenomena could lead to the model when approximated.

While descriptive and phenomenological models are important contributions to summarize processes, these models lack explanatory power. In contrast, mechanistic models allow researchers to extract information about the processes underlying the data. Mechanistic models, however, require an understanding about the underlying process which may not always be available. Even when this information is available, transition towards mechanistic models can be slow. For example, the most popular models used today to analyze codon usage are still phenomenological (Ikemura, 1981; Bennetzen and Hall, 1982; Sharp and Li, 1987; Wright, 1990; dos Reis *et al.*, 2003, 2004). While these phenomenological models provide good heuristics to explore differences in codon usage or other phenomena, they do not directly account for the evolutionary forces shaping the observed patterns such as selection, mutation, or genetic drift. Accounting for these forces allows for the proposal and testing of more sophisticated hypothesis as I demonstrate in chapter 3 and chapter 4.

5.1.2 Mechanistic Models Supplement Experiments

In addition to extracting information about biological processes from data, mechanistic models can help supplement experimental procedures. Empirical estimates of site specific selection are a valuable resource to e.g. identify sites conferring antibiotic resistance (Firnberg *et al.*, 2014; Stiffler *et al.*, 2016). While the unit that selection can act on is the amino acid, amino acids are a complex collection of physicochemical properties. It is, therefore, unclear for which properties amino acids are actually selected and when. Mechanistic models could be used to explore differences in the selection for physicochemical properties within and across proteins. Furthermore hypothesis could be formulated about the differing importance of physicochemical properties between e.g. sites or secondary structure elements

5.2 Estimating Protein Functional and Fitness Landscape

The selection on a protein sequence is highly complex. A protein of length L has 20^L possible states it can occupy in a L dimensional fitness landscape. This enormous complexity makes it prohibitively expensive to study protein fitness landscapes without simplifying assumptions. It is therefore important to be aware of potential impacts such assumptions have on the obtained results and how models can be further improved. However, despite such simplifying assumptions, valuable information has been extracted from protein coding sequences.

5.2.1 The Importance of Translation Errors

We often think of genes evolving with natural selection favoring proteins that encode their function optimally, with mutations and genetic drift reducing protein functionality. The error rate of protein synthesis is five to six orders of magnitude higher than mutations,

causing between 10% and 20% of average length proteins to contain errors (Goldsmith and Tawfik, 2009; Drummond and Wilke, 2009), creating more erroneous high expression proteins such as ribosomal proteins than error free low expression proteins. Selection on a gene is, therefore, not based solely on the error free protein sequence, but on the average fitness of the population of proteins resulting from a gene by means of error prone protein synthesis. Previous work showed that proteins with functionality essential to an organism can adapt to increased error rates by increasing gene expression and showed increased selection for more stable proteins (Goldsmith and Tawfik, 2009).

Organisms can take two routes to minimize the synthesis of proteins with altered functionality (Drummond and Wilke, 2009). First, organisms can evolve to minimize the rate at which errors during protein synthesis occur, e.g. selecting for codons that minimize translation error rates (Akashi, 2003; Gilchrist and Wagner, 2006). Second, selection could favor proteins with increased robustness to transcriptional and translational errors, e.g. increase protein stability or increase protein synthesis to compensate for non-functional proteins (Goldsmith and Tawfik, 2009).

In chapter 3, I assumed that the translation process is error free, and that each produced protein functions optimal. Thus, I explicitly ignore any selection on the reduction of translation error rates. While selection for the reduction of translation error rates and selection on ribosome overhead cost do not have to be counteracting forces, they could be for some synonymous codon families. The employed ROC SEMPPR framework (Gilchrist *et al.*, 2015) yields 100% usage of the most efficient codon if proteins synthesis rate is high enough. While individual genes may reach a 100% codon usage of the most efficient codon, we do not observe populations of high expression genes like that in nature. It is therefore unclear if selection for ribosome overhead cost can overpower counteracting selective forces if protein synthesis rate is high enough.

5.2.2 Homogeneous Selection

In ROC SEMPPR and *SelAC* functionality of a protein refers to the ability of a protein to perform its function and the overall need of an organism for the function. The functionality of a protein depends on many factors (Drummond and Wilke, 2009). As a result, we can approximate the functionality of the protein sequence $\vec{a}$ in a multitude of ways (Gibbs, 1873; Grantham, 1974; Cohen *et al.*, 2009). However, none can capture the full complexity of a folded protein. It is easy to imagine how the strength, or direction of selection can vary between amino acid site, secondary structures and protein domains within the same protein and certainly between proteins. For example, the functionality of a well-adapted protein is unlikely to be increased by an amino acid substitution. However, the effect on the functionality and in turn fitness of a substitution may drastically differ between active sites and structural sites. Similarly, the exchange of an hydrophilic amino acid with a hydrophobic amino acid is likely to have different effects on the surface of a protein than at the core.

The *SelAC* framework (Beaulieu *et al.*, 2019) employed in chapter 4 assumes that the efficacy of selection follows a gamma-distribution. This distribution is applied to all sites. I, therefore, explicitly do not account for potential differences in the distribution of selection between e.g. secondary structure elements. Similarly, selection for physicochemical properties may differ between sites in e.g. the core or at the surface of a protein. Improvements to *SelAC* with regards to these shortcomings would allow for new hypothesis to be tested and novel information to be gained from the same data.

Bibliography

Bibliography

Akashi, H., 2003 Translational selection and yeast proteome evolution. Genetics **164**: 1291–1303. 94

Akashi, H., and T. Gojobori, 2002 Metabolic efficiency and amino acid composition in the proteoms of *Escherichia coli* and *Bacillus subtilis*. Proccedings of the National Academy of Sciences U.S.A **99**: 3695–3670. 1

Altschul, S., 1991 Amino acid substitution matrices from an information theoretic perspective. Journal of Molecular Biology **219**: 555–565. 2

Anfinsen, C., 1973 Pronciples that govern the folding of protein chains. Science **181**: 223–230. 2

Ashenberg, O., L. Gong, and J. Bloom, 2013 Mutational effects on stability are largely conserved during protein evolution. Proccedings of the National Academy of Sciences U.S.A **110**: 21071–21076. 67

Ashkenazy, H., O. Penn, A. Doron-Faigenboim, O. Cohen, G. Cannarozzi, *et al.*, 2012 Fastml: a web server for probabilistic reconstruction of ancestral sequences. Nucleic Acids Research **40**: W580–4. 80

Baker, E., B. Wang, N. Bellora, D. Peris, A. Hulfachor, *et al.*, 2015 The genome sequence of saccharomyces eubayanus and the domestication of lager-brewing yeasts. Molecular Biology and Evolution **32**: 2818–2831. 48

Beaulieu, J., B. O'Meara, R. Zaretzki, C. Landerer, J. Chai, *et al.*, 2019 Population genetics based phylogenetics under stabilizing selection for an optimal amino acid sequence: A nested modeling approach. Molecular Biology and Evolution **NA**: NA. 1, 6, 67, 68, 70, 80, 95

Beimforde, C., K. Feldberg, S. Nylinder, J. Rikkinen, H. Tuovila, *et al.*, 2014 Estimating the phanerozoic history of the ascomycota lineages: combining fossil and molecular data. Mol. Phylogenet. Evol. **78**: 386–398. 38

Bennetzen, J., and B. Hall, 1982 Codon selection in yeast. J. Biol. Chem. **257**: 3026–3031. 2, 92

Bloom, J., 2014 An experimentally informed evolutionary model improves phylogenetic fit to divergent lactamase homologs. Molecular Biology and Evolution **31**: 2753–2769. 6, 68, 69

Bloom, J., 2017 Identification of positive selection in genes is greatly improved by using experimentally infromed site-specific models. Biology Direct **12**: 1. 6, 68, 69, 75, 79

Booch, G., 1993 *Object-oriented analysis and design with applications*. Benjamin-Cummings Publishing Co, Redwood City. 12

Brun, T., J. Peduzzi, M. Canica, G. Paul, P. Nevot, *et al.*, 1994 Characterization and amino acid sequence of irt-4, a novel tem-type enzyme with a decreased susceptibility to beta-lactamase inhibitors. FEMS Microbiology Letters **120**: 111–117. 69

Bulmer, M., 1990 The selection-mutation-drift theory of synonymous codon usage. Genetics **129**: 897–907. 37

Buttgereit, F., and M. Brand, 1995 A hierachy of atp-consuming processes in mammalian cells. Biochemical Journal **312**: 163–167. 1

Chanal, C., M. Poupart, D. Sirot, R. Labia, J. Sirot, *et al.*, 1992 Nucleotide sequences of caz-2, caz-6, and caz-7 beta-lactamase genes. Antimicrob. Agents Chemother. **36**: 1817–1820. 69

Cohen, M., V. Potapov, and G. Schreiber, 2009 Four distances between pairs of amino acids provide a precise description of their interaction. PLoS Comp. Biol. **5**: e1000470. 95

Cope, A., R. Hettich, and M. Gilchrist, 2018 Quantifying codon usage in signal peptides: Gene expression and amino acid usage explain apparent selection for inefficient codons. Biochimica et Biophysica Acta (BBA) - Biomembranes **1860**: 2479–2485. 49

Crow, J., and M. Kimura, 1970 *An introduction in Population Genetics Theory*. Harper and Row, 1649–1654. 77

Davis, M., and M. Pelsor, 2001 Experimental support for a resourcebased mechanistic model of invasibility. Ecology Letters **4**: 421–428. 2

Dayhoff, M., R. Schwartz, and B. Orcutt, 1978 A model of evolutionary change in proteins. Atlas of Protein Sequence and Structure **5**: 345–352. 2

Doron-Faigenboim, A., and T. Pupko, 2007 A combined empirical and mechanistic codon model. Molecular Biology and Evolution **24**: 388–397. 2

dos Reis, M., R. Savva, and L. Wernisch, 2004 Solving the riddle of codon usage preferences: a test for translational selection. Nucleic Acids Research **32**: 5036–5044. 38, 49, 92

dos Reis, M., L. Wernisch, and R. Savva, 2003 Unexpected correlations between gene expression and codon usage bias from microarray data for the whole escherichia coli k-12 genome. Nucleic Acids Research **31**: 6976–6985. 92

Drummond, D., and C. Wilke, 2008 Mistranslation-induced protein misfolding as a dominant constraint on coding-sequence evolution. Cell **134**: 341–352. 91

Drummond, D., and C. Wilke, 2009 The evolutionary consequences of erroneous protein synthesis. Nature Reviews **10**: 715–724. 94, 95

Dunn, C., F. Zapata, C. Munro, S. Siebert, and A. Hejnol, 2018 Pairwise comparisons across species are problematic when analyzing functional genomic data. Proc Natl Acad Sci USA **115**: E409–E417. 13

Echave, J., S. Spielman, and C. Wilke, 2016 Causes of evolutionary rate variation among protein sites. Nature Reviews Genetics **17**: 109–121. 67

Edelbuettel, D., and R. Francois, 2011 Rcpp: Seamless r and c++ integration. Journal of Statistical Software **40**: 1–18. 12, 22

Felsenstein, J., 1981 Evolutionary trees from dna sequences: a maximum likelihood approach. Journal of Molecular Evolution **17**: 368–376. 2, 67

Firnberg, E., J. Labonte, J. Gray, and M. Ostermeier, 2014 A comprehensive, high-resolution map of a gene's fitness landscape. Molecular Biology and Evolution **31**: 1581–1592. 77, 93

Fitch, W., 1976 Is there selection against wobble in codon-anticodon pairing? Science **194**: 1173–1174. 2

Friedrich, A., C. Reiser, G. Fischer, and J. Schacherer, 2015 Population genomics reveals chromosome-scale heterogeneous evolution in a protoploid yeast. Molecular Biology and Evolution **32**: 184 – 192. 4, 38, 43, 44, 47, 49

Gibbs, J., 1873 A method of geometrical representation of the thermodynamic properties of substances by means of surfaces. Transactions of the Connecticut Academy of Arts and Sciences **2**: 382–404. 95

Gibson, B., D. Wilson, E. Feil, and A. Eyre-Walker, 2018 The distribution of bacterial doubling times in the wild. Proc Biol Sci **285**: 20180789. 77

Gilchrist, M., 2007 Combining models of protein translation and population genetics to predict protein production rates from codon usage patterns. Molecular Biology and Evolution **24**: 2362–2372. 2, 3, 37, 50, 52, 80, 91

Gilchrist, M., W. Chen, P. Shah, C. Landerer, and R. Zaretzki, 2015 Estimating gene expression and codon-specific translational efficiencies, mutation biases, and selection coefficients from genomic data alone. Genome Biology and Evolution **7**: 1559–1579. 1, 2, 3, 4, 10, 13, 15, 31, 37, 38, 47, 50, 52, 91, 94

Gilchrist, M., P. Shah, and R. Zaretzki, 2009 Measuring and detecting molecular adaptation in codon usage against nonsense errors during protein translation. Genetics **183**: 1493–1505. 1, 91

Gilchrist, M., and A. Wagner, 2006 A model of protein translation including codon bias, nonsense errors, and ribosome recycling. J. of Theo. Biol. **239**: 417–434. 94

Gillespie, D., 1976 A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. Journal of Computational Physics **22**: 403–434. 79

Gojobori, T., 1983 Codon substitution in evolution and the "saturation" of synonymous changes. Genetics **105**: 1011–1027. 67

Goldman, N., and Z. H. Yang, 1994 Codon-based model of nucleotide substitution for protein-coding DNA-sequences. Molecular Biology and Evolution **11**: 725–736. 2, 6, 67

Goldsmith, M., and D. Tawfik, 2009 Potential role of phenotypic mutations in the evolution of protein expression and stability. Proccedings of the National Academy of Sciences U.S.A **106**: 6197–6202. 94

Goussard, S., W. Sougakoff, C. Mabilat, A. Bauernfeind, and P. Courvalin, 1991 An is1-like element is responsible for high-level synthesis of extended-spectrum beta-lactamase tem-6 in enterobacteriaceae. J. Gen. Microbiol. **137**: 2681–2687. 69

Gouy, M., and C. Gautier, 1982 Codon usage in bacteria: correlation with gene expressivity. Nucleic Acids Research **10**: 7055–7074. 37

Grantham, R., 1974 Amino acid differences formula to help explain protein evolution. Science **185**: 862–864. xi, 5, 79, 95

Grantham, R., C. Gautier, and M. Gouy, 1980 Codon frequencies in 119 individual genes confirms consistent choices of degenerate bases according to genome type. Nucleotide Acid Research **8**: 1893–1912. 2

Grantham, R., C. Gautier, M. Gouy, M. Jacobzone, and R. Mercier, 1981 Codon catalog usage is a genome strategy modulated for gene expressivity. Nucleic Acids Research **9**: 43–74. 2

Halpern, A., and W. Bruno, 1998 Evolutionary distances for protein-coding sequences: Modeling site-specific residue frequencies. Molecular Biology and Evolution **15**: 910–917. 67, 68

Hartl, D., E. Moriyama, and S. Sawyer, 1994 Selection intensity for codon bias. Genetics **138**: 227–234. 76

Henikoff, S., and J. Henikoff, 1992 Amino acid substitution matrices from protein blocks. Proceedings of the National Academy of Sciences of the United States of America **89**: 10925–10919. 67

Hilton, S., M. Doud, and J. Bloom, 2017 phydms: software for phylogenetic analyses informed by deep mutation scanning. PeerJ **5**: e3657. 68, 69, 77

Höhna, S., M. Landis, T. Heath, B. Boussau, N. Lartillot, *et al.*, 2016 Revbayes: Bayesian phylogenetic inference using graphical models and an interactive model-specification language. Systematic Biology **65**: 726–736. 67

Holder, M., D. Zwickl, and C. Dessimoz, 2008 Evaluating the robustness of phylogenetic methods to among-site variability in substitution processes. Philos Trans R Soc Lond B **363**: 4013–4021. 5

Hughes, A., 2007 Looking for darwin in all the wrong places: the misguided quest for positive selection at the nucleotide sequence level. Heredity **99**: 364–373. 67, 70

Hughes, A. L., and M. Nei, 1988 Pattern of nucleotide substitution at major histocompatibility complex class-i loci reveals overdominant selection. Nature **335**: 167–170. 67, 70

Ikemura, T., 1981 Correlation between the abundance of *Escherichia coli* transfer rnas and the occurrence of the respective codons in its protein genes: a proposal for a synonymous codon choice that is optimal for the *E. coli* translational system. Journal of Molecular Biology **151**: 389–409. 2, 92

Ikemura, T., 1985 Codon usage and trna content in unicellular and multicellular organisms. Molecular Biology and Evolution **2**: 13–34. 37

Jones, D., W. Taylor, and J. Thornton, 1992 The rapid generation of mutation data matrices from protein sequences. Bioinformatics **8**: 275–282. 67

Jukes, T., and C. Cantor, 1969 *Evolution of Protein Molecules*. Academic Press, 21–132. 2

Kensche, P., M. Oti, B. Dutilh, and M. Huynen, 2008 Conservation of divergent transcription in fungi. Trends Genet. **5**: 207–211. 48

Kimura, M., 1980 A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. Journal of Molecular Evolution **16**: 111–120. 2

Kosiol, C., I. Holmes, and N. Goldman, 2007 An empirical codon model for protein sequence evolution. Molecular Biology and Evolution **24**: 1464–1479. xiv, 71, 72

Lafay, B., P. Sharp, A. Lloyd, M. McLean, K. Devine, *et al.*, 1999 Proteome composition and codon usage in spirochaetes: Species-specific and dna strand-specific mutational biases. Nucleic Acids Research **27**: 1642–1649. 4

Landerer, C., A. Cope, R. Zaretzki, and M. A. Gilchrist, 2018 Anacoda: analyzing codon data with bayesian mixture models. Bioinformatics **34**: 2496–2498. 4, 39, 50

Lang, G. I., and A. W. Murray, 2008 Estimating the per-base-pair mutation rate in the yeast saccharomyces cerevisiae. Genetics **178**: 67 – 82. 51

Lartillot, N., and H. Philippe, 2004 A bayesian mixture model for across-site heterogeneities in the amino-acid replacement process. Molecular Biology and Evolution **21**: 1095–1109. 5, 68

Laureau, M., 1998 Biodiversity and ecosystem functioning: A mechanistic model. Proccedings of the National Academy of Sciences U.S.A **95**: 5632–5636. 2

Lawrence, J., and H. Ochman, 1997 Amelioration of bacterial genomes: Rates of change and exchange. Journal of Molecular Miology **44**: 383–397. 4, 37

Le, S., N. Lartillot, and G. O, 2008 Phylogenetic mixture models for proteins. Philos Trans R Soc Lond B Biol Sci **363**: 3965–3976. 5, 68

Leder, P., and M. Nierenberg, 1964 Rna codewords and protein synthesis, iii. on the nucleotide sequence of a cysteine and leucine rna codeword. Proccedings of the National Academy of Sciences U.S.A **52**: 1521–1529. 2

Lee, H., E. Popodi, H. Tang, and P. Foster, 2012 Rate and molecular spectrum of spontaneous mutations in the bacterium escherichia coli as determined by whole-genome sequencing. Proceedings of the National Academy of Sciences of the United States of America **109**: E2774–E2783. 76

Legendre, P., 2018 *lmodel2: Model II Regression*. R package version 1.7-3. 50

Liberles, D., A. Teufel, L. Liu, and T. Stadler, 2013 On the need for mechanistic models in computational genomics and metagenomics. Genome Biology and Evolution **5**: 2008–2018. 2

Lindqvist, L., K. Tandoc, I. Topisirovic, and L. Furic, 2018 Cross-talk between protein synthesis, energy metabolism and autophagy in cancer. Current Opinion in Genetics and Development **48**: 104–111. 1

Mabilat, C., J. Lourencao-Vital, S. Goussard, and P. Courvalin, 1992 A new example of physical linkage between tn1 and tn21: the antibiotic multiple-resistance region of plasmid pcff04 encoding extended-spectrum beta-lactamase tem-3. Mol Gen Genet **235**: 113–121. 69

Marcet-Houben, M., and T. Gabaldn, 2015 Beyond the whole-genome duplication: Phylogenetic evidence for an ancient interspecies hybridization in the baker's yeast lineage. PLoS Biology **13**: e1002220. 38, 48

Matthaei, J., and M. Nierenberg, 1961 Characteristics and stabilization of dnaase-sensitice protein synthesis in *E. coli* extracts. Proccedings of the National Academy of Sciences U.S.A **47**: 1580–1588. 1

Maxwell, E., 1962 Stimulation of amino acid incorporation into protein by natural and synthetic polyribonucleotides in a mammalian cell-free system. Proccedings of the National Academy of Sciences U.S.A **48**: 1639–1643. 2

McGill, B., R. Etienne, J. Gray, D. Alonso, M. Anderson, *et al.*, 2007 Species abundance distributions: moving beyond single prediction theories to integration within an ecological framework. Ecology Letters **10**: 995–1015. 2

Mdigue, C., T. Rouxel, P. Vigier, A. Hnaut, and A. Danchin, 1991 Evidence for horizontal gene transfer in escherichia coli speciation. Journal of Molecular Miology **222**: 851–856. 4, 37

Mi, G., Y. Di, and D. Schafer, 2015 Goodness-of-fit tests and model diagnostics for negative binomial regression of rna sequenceing data. PLOS ONE **10**: e0119254. 12

Muse, S., and B. Gaut, 1994 A likelihood approach for comparing synonymous and nonsynonymous nucleotide substitution rates, with application to the chloroplast genome. Molecular Biology and Evolution **11**: 715–724. 6

Neu, H., 1969 Effect of beta-lactamase location in escherichia coli on penicillin synergy. Appl Microbiol **17**: 783–786. 69

Nguyen, L., H. Schmidt, A. von Haeseler, and B. Minh, 2015 Iq-tree: A fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. Molecular Biology and Evolution **32**: 268–274. 67, 70, 79

Nierenberg, M., and J. Matthaei, 1961 The dependence of cell-free protein synthesis in *E. coli* upon naturally occurring or synthetic polyribonucleotides. Proccedings of the National Academy of Sciences U.S.A **47**: 1588–1602. 2

Nowak, M., 2006 *Evolutionary Dynamics: Exploring the Equations of Life*. Belknap of Harvard University Press. 67, 70

Ochman, H., and A. Wilson, 1987 *Evolutionary history of enteric bacterian*. ASM Press, 1649–1654. 76

O'Meara, B., C. Ane, M. Sanderson, and P. Wainwrigth, 2006 Testing for different rates of continuous trait evolution using likelihood. Evolution **5**: 922–933. 67

Payen, C., G. Fischer, C. Marck, C. Proux, D. J. Sherman, *et al.*, 2009 Unusual composition of a yeast chromosome arm is associated with its delayed replication. Genome Research **19**: 1710–1721. 38, 48, 49

R Core Team, 2015 *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. 10, 50, 79

Reeds, P., M. Fuller, and N. BA, 1985 *Metabolic basis of energy expenditure with particular reference to protein*. John Libby, 46–47. 1, 91

Rodrigue, N., and N. Lartillot, 2014 Site-heterogeneous mutation-selection models within the phylobayes-mpi package. Bioinformatics **30**: 1020–1021. 68

Rodrigue, N., N. Lartillot, and H. Philippe, 2008 Bayesian comparisons of codon substitution models. Genetics **180**: 1579–1591. 68

Romero, H., A. Zavala, and H. Musto, 2000 Codon usage in chlamydia trachomatis is the result of strand-specific mutational biases and a complex pattern of selective forces. Nucleic Acids Research **28**: 2084–2090. 4

Ruprecht, C., S. Proost, M. HernandezCoronado, C. OrtizRamirez, D. Lang, *et al.*, 2017 Phylogenomic analysis of gene coexpression networks reveals the evolution of functional modules. The Plant Journal **90**: 447–465. 67

Sagi, D., R. Rak, H. Gingold, I. Adir, G. Maayan, *et al.*, 2016 Tissue- and time-specific expression of otherwise identical trna genes. PLOS Genetics **12**: 1–27. 4

Schwartz, R., and A. Schäffer, 2017 The evolution of tumour phylogenetics: principles and practice. Nature Reviews Genetics **18**: 213–229. 67

Sella, G., and A. Hirsh, 2005 The application of statistical physics to evolutionary biology. Proceedings of the National Academy of Sciences of the United States of America **102**: 9541–9546. 48, 79

Shah, P., and M. Gilchrist, 2011a Explaining complex codon usage patterns with selection for translational efficiency, mutation bias, and genetic drift. Proccedings of the National Academy of Sciences U.S.A **108**: 10231–10236. 1, 2, 3, 37, 38, 91

Shah, P., and M. Gilchrist, 2011b Explaining complex codon usage patterns with selection for translational efficiency, mutation bias, and genetic drift. Proc Natl Acad Sci USA **108**: 10231–6. 10, 13

Sharp, P., E. Cowe, D. Higgins, D. Shields, K. Wolfe, *et al.*, 1988 Codon usage patterns in escherichia coli, bacillus subtilis, saccharomyces cerevisiae, schizosaccharomyces pombe, drosophila melanogaster and homo sapiens; a review of the considerable within species diversity. Nucleic Acids Research **16**: 8207–8211. 2

Sharp, P., and W. Li, 1987 The codon adaptation index - a meassure of directional synonymous codon usage bias, and its potential applications. Nucleic Acids Research **15**: 1281–1295. 2, 10, 38, 49, 92

Soderlund, C., M. Bomhoff, and W. Nelson, 2011 Symap v3.4: a turnkey synteny system with application to plant genomes. Nucleic Acids Research **39**: e68. 51

Soderlund, C., W. Nelson, A. Shoemaker, and A. Paterson, 2006 Symap A system for discovering and viewing syntenic regions of fpc maps. Genome Research **16**: 1159 – 1168. 50

Sokal, R., and F. Rohlf, 1981 *Biometry - The principles and practice of statistics in biological*. W. H. Freeman, 547–555. xii, xiii, 40, 41, 43, 50, 57, 58

Sougakoff, W., S. Goussard, and P. Courvalin, 1988 The tem-3 beta-lactamase, which hydrolyzes broad-spectrum cephalosporins, is derived from the tem-2 penicillinase by two amino acid substitutions. FEMS Microbiology Letters **56**: 343–348. 69

Sougakoff, W., A. Petit, S. Goussard, D. Sirot, A. Bure, *et al.*, 1989 Characterization of the plasmid genes blat-4 and blat-5 which encode the broad-spectrum beta-lactamases tem-4 and tem-5 in enterobacteriaceae. Gene **78**: 339–348. 69

Stamatakis, A., 2014 Raxml version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics **30**: 1312–1313. 67

Stiffler, M., D. Hekstra, and R. R, 2016 Evolvability as a function of purifying selection in tem-1 β-lactamase. Cell **160**: 882–892. 6, 69, 70, 79, 80, 93

Tamuri, A., N. Goldman, and M. dos Reis, 2014 A penalized likelihood method for estimating the distribution of selection coefficients from phylogenetic data. Genetics **197**: 257–271. 6

Tavare, S., 1986 Some probabilistic and statistical problems in the analysis of dna sequences. Lectures on Mathematics in the Life Sciences **17**: 57–86. 67

Thorne, J., N. Goldman, and D. Jones, 1996 Combinng protein evolution and secondary structure. Molecular Biology and Evolution **13**: 666–673. 6

Thyagarajan, B., and J. Bloom, 2014 The inherent mutational tolerance and antigenic evolvability of influenza hemagglutinin. eLife **3**: e03300. 6

Tsai, I., D. Bensasson, A. Burt, and V. Koufopanou, 2008 Population genomics of the wild yeast saccharomyces paradoxus: quantifying the life cycle. Proc Natl Acad Sci U.S.A. **105**: 4957–4962. 48

Tsankov, A., D. Thompson, A. Socha, A. Regev, and O. Rando, 2010 The role of nucleosome positioning in the evolution of gene regulation. PLoS Biol **8**: e1000414. xii, 40

Wagner, A., 2005 Energy constraints on the evolution of gene expression. Molecular Biology and Evolution **22**: 1365–1374. 48, 52

Wallace, E., E. Airoldi, and D. Drummond, 2013 Estimating selection on synonymous codon usage from noisy experimental data. Molecular Biology and Evolution **30**: 1438–1453. 10, 13, 37

Wang, H., K. Li, E. Susko, and A. Roger, 2008 A class frequency mixture model that adjusts for site-specific amino acid frequencies and improves inference of protein phylogeny. BMC Evolutionary Biology **8**: 331. 5

Warner, J., 1999 The economics of ribosome biosynthesis in yeast. Trends Biochem Sci **24**: 437–440. 1

Waterlow, J., and D. Millward, 1989 *Energy cost of turnover of protein and other cellular constituents*. Georg Thieme Verlag, 277–282. 1, 91

Whelan, S., and N. Goldman, 2001 A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. Molecular Biology and Evolution **18**: 691–699. 67

Wolfram Research Inc., 2017 *Mathematica 11*. 51

Wright, F., 1990 The 'effective number of codons' used in a gene. Gene **87**: 23–29. 2, 10, 26, 92

Wu, C., M. Suchard, and A. Drummond, 2013 Bayesian selection of nucleotide substitution models and their site assignments. Molecular Biology and Evolution **30**: 669–688. 6

Yang, S., and P. Bourne, 2009 The evolutionary history of protein domains viewed by species phylogeny. PLOS ONE **4**: e8378. 67

Yang, Z., 1994 Maximum-likelihood phylogenetic estimation from DNA-sequences with variable rates over sites - approximate methods. Journal Of Molecular Evolution **39**: 306–314. 67, 79

Zharkikh, A., 1994 Estimation of evolutionary distances between nucleotide sequences. Journal of Molecular Evolution **39**: 315–329. x, 70, 71

Zuckerkandl, E., and L. Pauling, 1962 *Molecular disease, evolution, and genic heterogeneity*. Academic Press, 189–225. 2

Vita

Cedric Landerer was born in Floersheim am Main, Germany on December 22, 1986 and raised in Frankfurt am Main, Germany. He graduated from Heinrich-Kleyer Highschool in Frankfurt, Germany in 2006. After that he moved to Munich, Germany to study Bioinformatics in a joined major at the University of Munich and Technical University, Munich. He recieved his Bachelor of Science in Bioinformatics in 2011 and his Masters of Science in Bioinformatics in 2013. He joined the Department of Ecology and Evolutionary Biology at the University of Tennessee, Knoxville in 2013 to persue his Ph. D. He recieved his Ph. D. in Ecology and Evolutionary Biology in December 2018 and will start a postdoctoral position at the Max Planck Institute for Molecular Cell Biology and Genetics in Dresden Germany in February 2019.