

Article

Methods for Parameter Estimation in Wine Fermentation Models

Robert Coleman ^{1,2,*} , James Nelson ¹  and Roger Boulton ¹ ¹ Department of Viticulture and Enology, University of California, Davis, CA 95616, USA; jjnel@ucdavis.edu (J.N.); rbboulton@ucdavis.edu (R.B.)² Department of Viticulture and Enology, Washington State University, Richland, WA 99164, USA

* Correspondence: bcoleman@ucdavis.edu

Abstract: The estimation of parameters in a wine fermentation model provides the opportunity to predict the rate and concentration outcomes, to strategically intervene to change the conditions, and to forecast the rates of heat and carbon dioxide release. The chosen parameters of the fermentation model are the initial assimilable nitrogen concentration and yeast properties (lag time, viability constant, and specific maintenance rate). This work evaluates six methods for parameter estimation: Bard, Bayesian Optimization, Particle Swarm Optimization, Differential Evolution, Genetic Evolution, and a modified Direct Grid Search technique. The benefits and drawbacks of the parameter computational methods are discussed, as well as a comparison of numerical integration methods (Euler, Runge–Kutta, backward differential formula (BDF), and Adams/BDF). A test set of density-time data for five white and five red commercial wine fermentations across vintage, grape cultivar, fermentation temperature, inoculated yeast strain, and fermentor size was used to evaluate the parameter estimation methods. A Canonical Variate Analysis shows that the estimation methods are not significantly different from each other while, in the parameter space, each of the fermentations were significantly different from each other.

Keywords: modeling; kinetics; parameter estimation; optimization; integration; computational methods



Citation: Coleman, R.; Nelson, J.; Boulton, R. Methods for Parameter Estimation in Wine Fermentation Models. *Fermentation* **2024**, *10*, 386. <https://doi.org/10.3390/fermentation10080386>

Academic Editor: Giuseppe Italo Francesco Perretti

Received: 19 June 2024

Revised: 20 July 2024

Accepted: 24 July 2024

Published: 27 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The need for monitoring, modeling, and controlling wine fermentations is of great economic importance and has been discussed in previous works [1–4]. Additionally, fermentation (sugar and yeast growth) models have been reviewed and evaluated to meet this need in both research and industry [3,5]. These models consist of sets of nonlinear ordinary differential equations (ODEs) requiring the use of numerical integration and parameter estimation/optimization. Equally, the solutions of the ODEs are not unique; however, as the model equations are based on the fundamental properties of yeast growth, sugar consumption, viability, and nitrogen uptake, the desired solution is determined by parameters bound to ranges of positive values which are supported by published values for *Saccharomyces cerevisiae* [6–8].

A variety of parameter estimation methods have previously been applied to fermentation models: Bard [9], Bayesian Optimization [10], Particle Swarm Optimization [11,12], Differential Evolution [11], Genetic Algorithm [11], and a modified Direct Grid Search technique [13]. These methods fall into general parameter estimation technique categories: gradient search (Bard), sequential design (Bayesian Optimization), bio-inspired or meta-heuristic (Particle Swarm Optimization, Differential Evolution, Genetic Algorithm), and brute force (modified Direct Grid Search). Each method and family have their advantages and disadvantages: the ability to find global minima, computer memory space, computational speed, and ease of cloud-based and/or parallelization of computer operation.

With the exception of the Bard method, the integration of the fermentation model ODEs requires a separate (external to parameter estimation) integration method to compute the resulting predicted sugar consumption concentrations. The Euler method, with a

uniform time step of an hour, was previously used with the Boulton fermentation model [9]. Alternative higher-order integration formulas, such as the Runge–Kutta, backward differential formula (BDF), and Adams/BDF, sometimes offer superior stability and accuracy. These methods are easily implemented and available through the Python SciPy library [14].

This work will evaluate these alternative estimation methods, as well as provide an initial optimization of the Euler step function when it is applied to the analysis of fermentation patterns. The parameter estimation methods will be compared based on their ability to fit the Boulton model [5,9] against 10 commercial fermentations ranging across vintage, grape cultivar, fermentation temperature, starter yeast, and fermentation volume.

2. Materials and Methods

2.1. Fermentations

Ten commercial fermentations were chosen for this analysis from the 2018, 2019, and 2020 harvest seasons (Treasury Wine Estates, Saint Helena). The fermentations were selected to include an extensive representation varying in vintage, grape cultivar, fermentation temperature, starter yeast, and fermentor size. The attributes of the fermentations are listed in Table 1, and the °Brix–time courses are plotted in Figure 1. The complete datasets for the ten fermentations, including time, temperature, and °Brix, are published in the Supplementary Materials (Table S1).

Table 1. Attributes of the ten commercial fermentations.

ID	Vintage	Cultivar	Inoculum (g/L)	Temperature (°C) ¹	Yeast	Volume (kL)
417A	2018	Chardonnay	0.258	18.95	VL1	9.16
417B	2019	Riesling	0.243	13.85	VL1	5.43
450	2020	Cabernet Sauvignon	0.247	25.85	BDX	11.30
457	2020	Cabernet Sauvignon	0.242	29.15	BO213	11.53
486	2020	Syrah	0.321	27.25	M2	9.82
489	2020	Cabernet Sauvignon	0.243	27.95	FX10	13.57
571	2019	Sauvignon blanc	0.240	13.35	VL3	4.54
823	2020	Sauvignon blanc	0.270	14.25	VL3	4.38
826	2020	Pinot noir	0.211	24.65	RC212	2.55
829	2019	Muscat	0.265	13.65	VL3	4.11

¹ An average temperature (isothermal) was used in the parameter optimization analysis. The temperature data are available in the Supplementary Materials (Table S1).

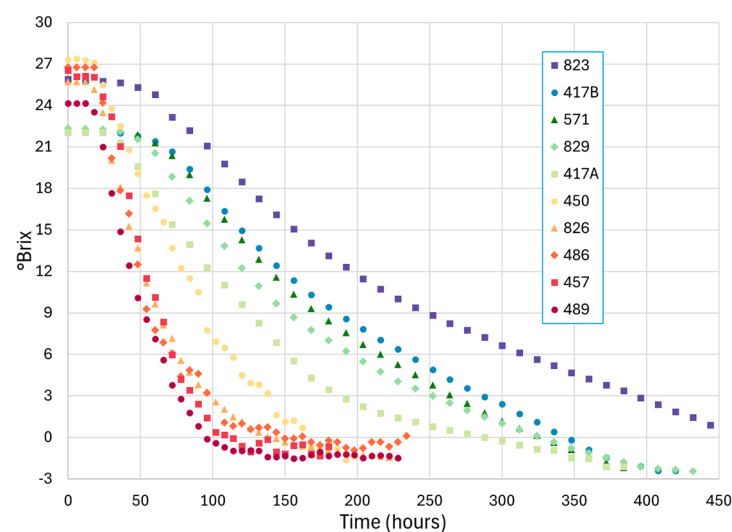


Figure 1. Brix–time plots of ten commercial fermentations. Time at 0 h is the point of yeast inoculation.

2.2. Fermentation Model

The density form of the Boulton model was used to evaluate the ten commercial fermentations; this is described in detail elsewhere [5]. The model consists of eight ODEs that are solved using a numerical integration method (i.e., Euler). Four enological parameters (lag time (h), initial nitrogen (mg/L), specific maintenance rate (1/h), and viability constant (L/g/h)) are optimized/fitted against available measurements from the fermentation: sugar consumption time course data (°Brix versus time in hours for this work), fermentation temperature (isothermal for this work), and initial yeast inoculum (g/L). The optimized parameter estimates are returned to the model and integrated to provide the ‘predicted’ fermentation sugar consumption curve (generally known as the “Brix curve”). It should be noted that the Boulton model includes a parameter for ethanol inhibition; however, in this work, this parameter was held constant at 0.035 based on the previous and extensive use of the model and the insensitivity of this parameter’s impact on the final prediction outcome.

2.3. Parameter Estimation Methods

Six parameter estimation methods were compared: Bard (B), Bayesian Optimization with a Sequential Monte Carlo sampler (BO-SMC), Particle Swarm Optimization (PSO), Differential Evolution (DE), Genetic Algorithm (GA), and a modified Direct Grid Search (MDGS). The Bard method, using the Davidon–Fletcher–Powell [15] approach of derivative estimation, is run as a standalone executable software program originally compiled in Fortran [16,17]. The MDGS method runs as Python-based code for this analysis [13]. The remaining methods are run in Python from available statistical packages or online published code: BO-SMC from the ‘PyMC’ library [18], PSO from online code [19], DE from the ‘limfit’ package [20], and GA from the ‘genetic algorithm’ package [21].

As needed per the method type, the methods were provided with an initial guess of the four parameters required by the Boulton model. Additionally, they were all given bounds or extents of the enological parameters based on the fundamental knowledge and experimentation of said parameters (see Supplementary Material—Table S2). The use and value of fundamental parameters was the main reason why machine learning or Artificial Intelligence optimization approaches were not needed in this work.

The Bard method is unique in this set of tested methods as it uses a gradient-based technique for parameter estimation. It is therefore prone to finding local minima or failing to converge based on initial guesses. To overcome this problem, a simple grid search was employed to seed the Bard method with initial guesses of low, middle, and high values based on the given bounds of each parameter. This resulted in running the method $3^4 = 81$ times while evaluating for the best overall fit. The best fit for all methods was determined by minimizing the residuals between the predicted and actual measurements using the residual sum of squares (RSS).

The parameter estimation methods were not optimized for their runtime performance as the goal of this work was to compare the accuracy and precision of methods, not computational performance. The method-, library-, or package-specific options are provided in the Supplementary Materials (Table S3).

2.4. Numerical Integration Methods

The numerical integration methods employed for each of the parameter estimation methods (except for the Bard method) include Euler and those found in the SciPy library: RK45, RK23, DOP853, Radau, BDF, and LSODA [14]. Unlike Euler, these numerical integration methods in the SciPy library are adaptive; therefore, a separate analysis of Euler was performed to provide an initial optimization of step size.

Euler step sizes of 1, 2, 5, 15, 30, 60, 120, and 300 min were tested using the fermentation ID ‘450’ with average parameter estimates from the DE optimization method. The 1 min step size was used as a reference to cross-plot against the remaining larger step sizes. Additionally, the root mean square error (RMSE), mean absolute error (MAE), and percent relative absolute error (%RAE) were calculated against the 1 min step size.

The cross comparison of the numerical integration methods was formulated using a 1 h (60 min) Euler step size against the remaining SciPy methods. The RMSE, MAE, and %RAE were calculated for the predicted sugar consumption curves of fermentation ID '450' as an example. The complete set of fermentation results are summarized in the Supplementary Materials (Table S4).

2.5. Comparison of Parameter Estimation Methods

The measured sugar consumption curves were fitted by each parameter estimation method with the means, standard deviations, and RMSE calculated. To determine statistical significance, one-way MANOVAs were individually carried out on the factors 'parameter estimation methods' and 'Fermentation ID' (fermentations). Canonical Variate Analysis (CVA) was completed to provide further statistical insight. Comparison analyses were carried out in R [22] and RStudio [23] with the 'candisc' package [24].

3. Results

3.1. Analysis of Euler Integration Step Size

To assess the error caused by varying interval step sizes in the Euler integration method, a small uniform step size of 1 min was selected as a benchmark. The resulting paired values ($^{\circ}\text{Brix}$ of 1 min step size versus incrementing step sizes) were evaluated on a cross plot shown in Figure 2A. Step sizes from 5 to 120 min showed no visual change in the 1-to-1 linearity through the $x = y$ line. At a step size of 300 min, the result showed a deviation of the $x = y$ line, indicating a significant truncation error in the integration. To further validate this visual indicator, RMSE, MAE, and %RAE were calculated for each step size against the 1 min step size (Table 2). The %RAE showed a 5.61% error in the 300 min step size; therefore, if a 5% error cutoff is used, one can use up to at least 120 min with acceptable outcomes. Previous unpublished works by the authors have indicated that a 1 h step size has the benefit of minimal error and desirable computational efficiency.

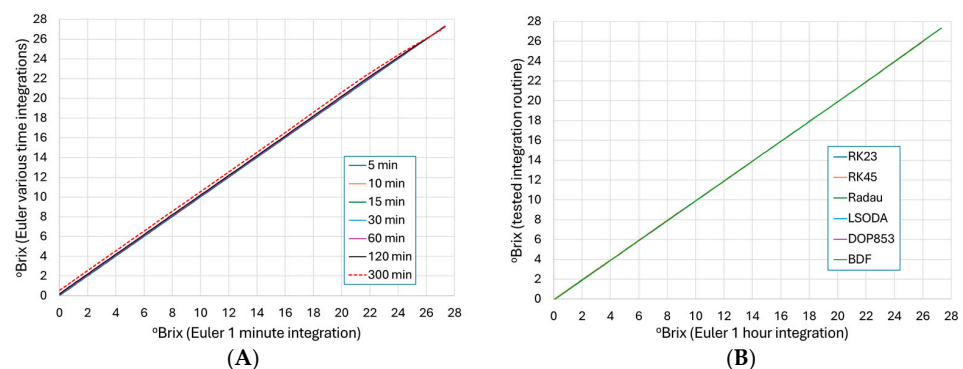


Figure 2. Cross plots of numerical integration methods using fermentation ID '450' and the average parameter estimates optimized by the Differential Evolution method. (A) Integrated $^{\circ}\text{Brix}$ values at a Euler step size of 1 min versus Euler step sizes of 5, 10, 15, 30, 60, 120, or 300 min; (B) Comparison of $^{\circ}\text{Brix}$ at a 1 h (60 min) Euler step size versus the integration methods from the SciPy library. There was no observed difference in the Euler step sizes from 5 to 120 min; also, there was no observed difference in the 6 integration methods compared to a 60 min Euler.

Table 2. Statistical analysis of incrementing step sizes on the Euler integration method.

	Euler Integration Step Size (minutes)						
	5	10	15	30	60	120	300
RSME	0.006	0.014	0.022	0.046	0.095	0.195	0.513
MAE	0.006	0.014	0.021	0.044	0.091	0.186	0.489
%RAE	0.07	0.16	0.25	0.15	1.05	2.16	5.61

3.2. Analysis of Alternate Integration Methods

To evaluate the performance of alternate numeric integration methods found in the SciPy library, a cross plot of the Euler integration with a step size of 1 h was graphed against the various alternative methods (Figure 2B). Visually, there appears to be no deviation in the methods from the $x = y$ line. Further statistical analysis (Table 3) indicates that the methods' %RAE were ~1% or less; therefore, it was concluded that any of the numerical integration methods are suitable for the Boulton model for these commercial fermentations. Any subsequent analysis in this work utilized the Euler method with a 1 h integration step size due to its ease of implementation and the convenience of producing whole number hourly outputs.

Table 3. Statistical analysis of alternative integration methods versus a 1 h Euler step size.

	Numerical Integration Method					
	RK23	RK45	Radau	LSODA	DOP853	BDF
RSME	0.102	0.066	0.092	0.097	0.094	0.096
MAE	0.097	0.057	0.088	0.093	0.090	0.092
%RAE	1.13	0.66	1.02	1.08	1.05	1.07

3.3. Analysis of Parameter Estimation Methods

The RMSE analysis summarized in Table 4 indicates that the magnitude of error between parameter estimation methods was less than the magnitude of error between fermentations. The discrepancy between fermentations was anticipated due to the noise in the °Brix–time course measurements in a commercial winery setting. This noise is visually apparent in Figure 1, with fermentation ID '486' being especially noted at the end of the fermentation. Conversely, the relatively consistent error between parameter estimation methods within a single fermentation provides us with confidence that the methods are fitting the curves similarly.

Table 4. Root mean squared error by method for each fermentation.

ID:	417A	417B	450	457	486	489	571	823	826	829
Root Mean Squared Error (RMSE)										
B	0.481	0.079	0.446	0.314	0.516	0.268	0.077	0.240	0.338	0.104
BO-SMC	0.499	0.167	0.487	0.324	0.515	0.312	0.090	0.252	0.338	0.140
DE	0.477	0.112	0.456	0.314	0.510	0.303	0.082	0.241	0.326	0.137
GA	0.479	0.129	0.485	0.328	0.512	0.306	0.090	0.243	0.330	0.144
PSO	0.480	0.129	0.469	0.318	0.523	0.309	0.115	0.260	0.361	0.161
MDGS	0.487	0.151	0.500	0.350	0.555	0.363	0.131	0.270	0.373	0.181
mean	0.484	0.128	0.474	0.325	0.522	0.310	0.098	0.251	0.344	0.145

A more detailed analysis of the methods and the resulting parameter estimates is shown in the means and %RSD in Table 5. Although the RMSE of each fermentation is of similar magnitude, the parameter estimates vary between methods, indicating multiple solutions or optimization minima for the model fitting. This was to be expected given the inherent nature of and criteria used by each parameter estimation method to search for the minimal residual error.

To determine if the parameter estimation methods were significantly different in their estimates, a MANOVA was performed with the parameter estimation method as the factor ($p \leq 0.05$). The resulting MANOVA (Supplementary Material—Table S5) was insignificant; therefore, the null hypothesis that the estimates provided by parameter estimation methods are equal across fermentations could not be rejected. This indicates that the parameter estimation methods are using the Boulton model parameters in a similar way to fit the measured fermentation curves.

Table 5. Means table with percent relative standard deviation (%RSD) for parameter estimates by method for each fermentation ID.

Lag Time (hours)										
ID:	417A	417B	450	457	486	489	571	823	826	829
Method	Mean ± %RSD									
B	41.7 ± 12.5	68.0 ± 1.9	13.1 ± 3.7	27.9 ± 3.9	20.9 ± 2.4	12.9 ± 5.0	62.6 ± 1.6	63.4 ± 5.2	16.8 ± 6.4	46.9 ± 2.2
BO-SMC	35.2 ± 0.3	63.2 ± 2.0	15.3 ± 8.7	25.6 ± 1.5	18.5 ± 0.9	10.8 ± 3.6	62.9 ± 2.2	58.1 ± 0.5	7.6 ± 9.9	50.3 ± 0.0
DE	35.7 ± 0.0	72.0 ± 0.0	11.3 ± 0.0	25.3 ± 0.0	18.8 ± 0.0	9.8 ± 0.0	65.2 ± 0.0	63.3 ± 0.0	6.5 ± 0.0	50.5 ± 0.3
GA	33.7 ± 8.1	69.2 ± 3.6	14.9 ± 19.7	26.8 ± 4.2	18.7 ± 2.7	9.3 ± 4.7	64.2 ± 3.8	61.0 ± 2.6	6.4 ± 14.7	52.8 ± 1.9
PSO	36.4 ± 3.0	70.2 ± 4.4	10.6 ± 10.9	25.8 ± 3.1	20.1 ± 0.9	10.3 ± 8.7	69.3 ± 6.7	64.5 ± 10.2	4.6 ± 87.3	51.8 ± 6.6
MDGS	36.3 ± 10.8	68.1 ± 3.0	13.3 ± 24.4	25.9 ± 6.3	19.8 ± 3.7	10.5 ± 23.4	64.9 ± 4.7	56.4 ± 3.7	7.4 ± 38.5	49.9 ± 10.9
Initial Nitrogen (mg/L)										
ID:	417A	417B	450	457	486	489	571	823	826	829
Method	Mean ± %RSD									
B	131 ± 9.3	158 ± 1.0	126 ± 0.2	209 ± 4.5	320 ± 0.6	214 ± 2.4	164 ± 1.3	145 ± 2.4	238 ± 3.4	189 ± 0.7
BO-SMC	144 ± 0.5	136 ± 1.2	149 ± 4.3	230 ± 2.2	327 ± 0.7	243 ± 1.9	156 ± 1.9	135 ± 0.3	305 ± 2.9	182 ± 0.0
DE	145 ± 0.0	150 ± 0.0	132 ± 0.0	226 ± 0.0	333 ± 0.0	233 ± 0.0	161 ± 0.1	141 ± 0.0	294 ± 0.0	182 ± 0.3
GA	139 ± 5.1	143 ± 3.3	148 ± 8.7	244 ± 6.3	332 ± 2.2	230 ± 1.4	158 ± 3.8	137 ± 2.3	291 ± 3.4	188 ± 1.2
PSO	143 ± 1.5	145 ± 4.3	133 ± 0.4	232 ± 4.8	350 ± 0.0	235 ± 1.9	169 ± 6.0	143 ± 7.1	282 ± 7.8	185 ± 6.2
MDGS	150 ± 5.8	145 ± 1.9	139 ± 6.3	235 ± 9.3	322 ± 3.5	227 ± 7.7	161 ± 5.3	136 ± 4.4	293 ± 6.6	184 ± 4.6
Specific Maintenance Rate (1/h)										
ID:	417A	417B	450	457	486	489	571	823	826	829
Method	Mean ± %RSD									
B	0.169 ± 13.7	0.129 ± 2.0	0.300 ± 0.6	0.224 ± 8.8	0.081 ± 1.9	0.200 ± 4.9	0.144 ± 2.7	0.143 ± 6.0	0.126 ± 7.1	0.121 ± 1.7
BO-SMC	0.175 ± 1.1	0.159 ± 2.0	0.256 ± 6.4	0.217 ± 4.2	0.084 ± 2.3	0.180 ± 4.4	0.154 ± 3.5	0.153 ± 0.4	0.101 ± 8.3	0.124 ± 0.1
DE	0.171 ± 0.0	0.133 ± 0.1	0.300 ± 0.0	0.221 ± 0.0	0.079 ± 0.0	0.196 ± 0.0	0.146 ± 0.4	0.143 ± 0.0	0.111 ± 0.0	0.124 ± 0.6
GA	0.183 ± 8.0	0.147 ± 5.7	0.261 ± 11.3	0.192 ± 12.7	0.079 ± 6.9	0.198 ± 2.1	0.151 ± 7.8	0.150 ± 3.9	0.115 ± 8.8	0.116 ± 2.4
PSO	0.176 ± 2.9	0.144 ± 9.0	0.300 ± 0.0	0.210 ± 9.6	0.067 ± 1.8	0.194 ± 1.7	0.134 ± 12.2	0.140 ± 10.7	0.123 ± 19.2	0.122 ± 13.2
MDGS	0.162 ± 10.1	0.141 ± 4.0	0.281 ± 7.3	0.207 ± 19.2	0.091 ± 16.9	0.209 ± 17.3	0.142 ± 11.0	0.149 ± 9.1	0.114 ± 17.8	0.120 ± 9.9

Table 5. Cont.

Viability Constant (L/g/h)										
ID:	417A	417B	450	457	486	489	571	823	826	829
Method	Mean ± %RSD									
B	20.2 ± 3.2	29.3 ± 2.9	27.9 ± 0.4	19.3 ± 2.6	22.4 ± 1.1	18.5 ± 1.9	18.3 ± 2.9	23.7 ± 4.7	25.8 ± 2.0	15.8 ± 1.8
BO-SMC	17.4 ± 0.3	25.0 ± 2.4	26.4 ± 0.2	17.9 ± 0.2	21.4 ± 0.6	17.1 ± 0.3	18.3 ± 2.0	23.8 ± 0.7	20.0 ± 1.8	16.6 ± 0.1
DE	17.3 ± 0.0	29.3 ± 0.1	26.5 ± 0.0	17.9 ± 0.0	21.7 ± 0.0	17.0 ± 0.0	18.8 ± 0.9	23.8 ± 0.0	19.6 ± 0.0	16.5 ± 0.6
GA	17.2 ± 0.4	25.6 ± 4.4	26.3 ± 0.4	18.0 ± 1.8	21.8 ± 1.5	17.2 ± 0.9	18.6 ± 6.7	23.3 ± 2.4	19.4 ± 1.5	17.0 ± 1.6
PSO	17.0 ± 1.8	26.0 ± 12.2	26.4 ± 0.7	18.0 ± 1.3	22.4 ± 1.8	16.8 ± 2.1	20.8 ± 9.9	24.8 ± 2.2	19.7 ± 2.6	16.6 ± 9.1
MDGS	17.5 ± 3.5	28.8 ± 3.8	26.6 ± 2.3	18.2 ± 3.9	20.7 ± 8.7	16.8 ± 5.0	21.6 ± 9.9	24.5 ± 11.7	19.4 ± 3.1	17.9 ± 6.3

Given the similarity in the resulting parameter estimates, a review of the means table (Table 5) shows that the DE method often had the lowest relative standard deviation, indicating that it was the method with the highest precision. Additionally, the DE routine had the lowest RMSE or one of the lowest RMSEs in each fermentation, thus indicating its higher accuracy.

3.4. Analysis of Fermentations

The difference in the rates and patterns between the fermentations are apparent in Figure 1. A MANOVA with 'Fermentation ID' as the factor (Supplementary Material—Table S6) was significant ($p \leq 0.05$), therefore indicating that the parameter estimates of the fermentations were significantly different. Thus, the MANOVA model was used in the CVA and graphed in Figure 3. The first two dimensions of these CVA diagrams (Figure 3A,B) account for 92.4% of the variance ratio within this dataset. The horizontal separation to the right hemisphere is associated with longer lag times in the white fermentations, primarily due to their cooler temperature conditions. The warmer red fermentations and those with higher levels of initial yeast assimilable nitrogen occupy the left hemisphere. The vertical separation is due to the higher maintenance rates and higher viability constants in the lower hemisphere and higher initial assimilable nitrogen in the upper hemisphere. It is interesting to note that the fermentations with higher initial nitrogen levels are also associated in this space with those with lower maintenance rates and viability constants, the two parameters that distinguish the tailing pattern of slow and incomplete fermentations. This pattern has been observed in the more extensive model analysis of commercial fermentations [25], where the enhancement of growth and growth rate by higher assimilable nitrogen levels does not always carry through into more active non-growing cells and higher cell viability.

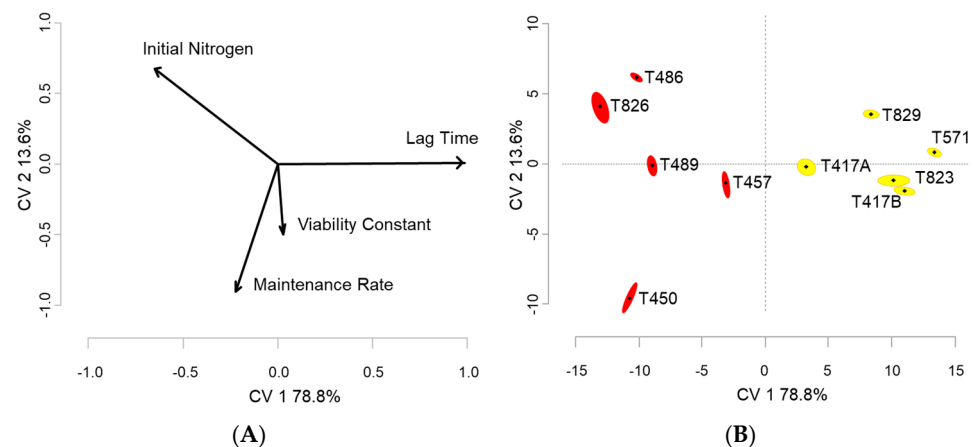


Figure 3. The CVA of parameter estimations effects' on the ten fermentations: (A) Loadings plot of parameter estimates; (B) Score plots of the 10 fermentations.

Regardless of the parameter estimation routines considered, the separation of these ten fermentations based on their parameter values (Figure 3A,B) indicates that the density version of the model ($^{\circ}$ Brix consumption) [3] can describe their dynamic nature effectively across volume, cultivar, yeast strain, inoculum level, fermentation temperature, and initial composition. This model is unique in its description of non-growing yeast activity and cell viability in the second stage of the fermentations, and this description is confirmed by the success of its application to the fermentation patterns and the parameter estimation in this diverse set of wine fermentations.

4. Discussion

While several alternative parameter estimation methods exist, the large batch-to-batch variation observed in wine fermentations can make online parameter estimation challenging. In general, parameter estimation techniques for nonlinear systems can be

separated into gradient methods, sequential design, bio-inspired algorithms, and computational/brute force. In gradient methods, such as Bard, the gradient of the objective function is calculated at each evaluation [16]. The benefit of gradient methods is that the standard deviation for each parameter can be approximated from the Hessian matrix at the last iteration. The standard deviation provides a confidence interval for the solution of each parameter. The performance of gradient methods is dependent on whether the initial guess is close to the global solution. In its application to wine fermentation, there is typically not enough information to select good initial guesses, sometimes requiring multiple iterations to find the global solution.

In bio-inspired algorithms such as PSO, GA, and DE, only lower and upper bounds are provided, making these algorithms attractive for automated parameter estimation across concurrent fermentations; however, these algorithms typically suffer from long computation times. While long computation times are not generally a problem for a single fermentation, at the winery scale, long computation times make cloud implementation more complex as tasks must be queued and executed asynchronously instead of being performed on demand.

In Bayesian estimation methods such as BO-SMC, the system model, measurement model, state noise, and measurement noise are determined a priori [26]. When estimating the unknown parameters, it is necessary to account for the state and measurement noises, which are often computationally intensive and a source of convergence difficulties [26].

Brute force methods such as MDGS are typically limited in the range and number of parameters; however, the computations can be performed in parallel across processors. The results in this work show that all parameter estimation methods converged to similar solutions; therefore, a method should be selected for its ability to handle large batch-to-batch variation and its ease of implementation.

Machine learning has also been used to develop models and predictions of biological processes where the underlying mechanics are unknown [27–30]; however, the coefficients and biases generally have no relationship or meaning concerning the underlying process, making the interpretation of results less meaningful. For this reason, these approaches were not considered in this work.

The lack of significant differences between the parameter estimation methods and integration routines suggest that the focus should be on improving density measurements (i.e., density sensors, methods, and placement), as well as improving first principle model descriptions of non-growing cells' activity and their decline. Given the slow rates of change in wine fermentations, the speed of the parameter estimation method is generally not a limitation. What is important is that the parameter estimation method should be robust and require little or no human intervention to adjust the bounds or initial guesses. In turn, the selection of a parameter estimation method becomes a choice of the user's ultimate need: robust/no human intervention, memory space, and/or processor time or computing efficiency.

5. Conclusions

All the parameter estimation methods considered produced satisfactory parameter values. There was no significant difference between the methods in the parameter space of a diverse fermentation dataset. In contrast, each fermentation was significantly different from the others in the parameter space.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/fermentation10080386/s1>, Table S1: Ten commercial fermentations °Brix, temperature and time data; Table S2: Parameter bounds; Table S3: Method-, library-, or package-specific options; Table S4: Numerical integration comparison across methods and fermentations; Table S5: MANOVA for estimated parameters and parameter optimization methods; Table S6: MANOVA for estimated parameters and ten commercial fermentations.

Author Contributions: Conceptualization, R.C., J.N. and R.B.; methodology, R.C., J.N. and R.B.; software, R.C., J.N. and R.B.; validation, R.C., J.N. and R.B.; formal analysis, R.C., J.N. and R.B.; investigation, R.C., J.N. and R.B.; data curation, R.C.; writing—original draft preparation, R.C.; writing—review and editing, R.C., J.N. and R.B.; visualization, R.C.; supervision, R.B. All authors have read and agreed to the published version of the manuscript.

Funding: The Stephen Sinclair Scott Endowment (R.B.) and Rodgers University Fellowship in Electrical and Computer Engineering (J.N.).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are available upon request.

Acknowledgments: Treasury Wine Estates, Saint Helena, CA.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Knoesen, A.; Boulton, R. A Brief History of the Modeling, Control and Optimization of Wine. *Fermentation* 2024, *in submission*.
2. Boulton, R.B.; Singleton, V.L.; Bisson, L.F.; Kunkee, R.E. *Principles and Practices of Winemaking*; Chapman & Hall: New York, NY, USA; Springer: Boston, MA, USA, 1996; ISBN 0-412-06411-1. [\[CrossRef\]](#)
3. Marín, M.R. Alcoholic Fermentation Modelling: Current State and Perspectives. *Am. J. Enol. Vitic.* **1999**, *50*, 166–178. [\[CrossRef\]](#)
4. Bisson, L.F.; Butzke, C.E. Diagnosis and Rectification of Stuck and Sluggish Fermentations. *Am. J. Enol. Vitic.* **2000**, *51*, 168–177. [\[CrossRef\]](#)
5. Nelson, J.; Boulton, R. Models for Wine Fermentation and Their Suitability for Commercial Applications. *Fermentation* **2024**, *10*, 269. [\[CrossRef\]](#)
6. Pirt, S.J. *Principles of Microbe and Cell Cultivation*; Wiley: New York, NY, USA, 1975; ISBN 0470690380/9780470690383.
7. Aiba, S.; Humphrey, A.E.; Millis, N.F. *Biochemical Engineering*, 2nd ed.; Academic Press: New York, NY, USA, 1973; ISBN 0-12-045052-1.
8. Bailey, J.E. *Biochemical Engineering Fundamentals*; McGraw Hill: New York, NY, USA, 1977; ISBN 0-07-003210-6.
9. Boulton, R. The Prediction of Fermentation Behavior by a Kinetic Model. *Am. J. Enol. Vitic.* **1980**, *31*, 40–45. [\[CrossRef\]](#)
10. Coleman, M.C.; Fish, R.; Block, D.E. Temperature-Dependent Kinetic Model for Nitrogen-Limited Wine Fermentations. *Appl. Environ. Microbiol.* **2007**, *73*, 5875–5884. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Coleman, R.E.; Boulton, R.B. Alternative Estimation Routines for Modeling and Prediction of Commercial Wine Fermentations. In Proceedings of the 72nd American Society of Enology and Viticulture National Conference, Virtual, 24 June 2021.
12. AWRI. Fermentation Simulator. Available online: https://www.awri.com.au/industry_support/winemaking_resources/wine_fermentation/awri-fermentation-simulator/ (accessed on 18 May 2024).
13. Nelson, J.N. The Digitization of Wine Fermentation. Ph.D. Thesis, University of California Davis, Davis, CA, USA, 2023.
14. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. Scipy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Fletcher, R.; Powell, M.J.D. A Rapidly Convergent Descent Method for Minimization. *Comput. J.* **1963**, *6*, 163–168. [\[CrossRef\]](#)
16. Bard, Y. Comparison of Gradient Methods for the Solution of Nonlinear Parameter Estimation Problems. *SIAM J. Numer. Anal.* **1970**, *7*, 157–186. [\[CrossRef\]](#)
17. Bard, Y. *Nonlinear Parameter Estimation*; Academic Press: New York, NY, USA, 1974; ISBN 0-12-078250-2.
18. Abril-Pla, O.; Andreani, V.; Carroll, C.; Dong, L.; Fonnesbeck, C.J.; Kochurov, M.; Kumar, R.; Lao, J.; Luhmann, C.C.; Martin, O.A.; et al. Pymc: A Modern, and Comprehensive Probabilistic Programming Framework in Python. *PeerJ Comput. Sci.* **2023**, *9*, e1516. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Rahmadya. Particle Swarm. Available online: <https://rahmadya.com/2020/05/31/particle-swarm-optimization-in-jupyter-notebook/> (accessed on 18 May 2024).
20. Newville, M.; Stensitzki, T.; Allen, D.B.; Ingargiola, A. Lmfit: Non-Linear Least-Square Minimization and Curve-Fitting for Python (0.8.0). Zendo. 2014. Available online: <https://zenodo.org/records/10998841> (accessed on 11 February 2024).
21. Python Package Index—Pypi. Python Software Foundation. Available online: <https://pypi.org/> (accessed on 12 February 2024).
22. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2023. Available online: <https://www.R-project.org/> (accessed on 1 December 2023).
23. RStudio Team. *Rstudio: Integrated Development Environment for R*; RStudio, PBC: Boston, MA, USA, 2023; Available online: <http://www.rstudio.com/> (accessed on 1 December 2023).
24. Friendly, M.; Fox, J. Candisc: Visualization Generalized Canonical Correlation Analysis. R. Package Version 0.9.0. 2024. Available online: <https://CRAN.R-project.org/package=heplots> (accessed on 6 November 2021).

25. Nelson, J.; Coleman, R.; Gravesen, P.; Silacci, M.; Velasquez, A.; Marinell, K. Analysis of a Commercial Red Wine Fermentation Dataset with a Wine Kinetic Model. *Fermentation* 2024, *in submission*.
26. Shyam, M.M.; Naik, N.; Gemson, R.M.O.; Ananthasayanam, M.R. Introduction to the Kalman Filter and Tuning Its Statistics for near Optimal Estimates and Cramer Rao Bound. *arXiv* **2015**, arXiv:1503.04313. [[CrossRef](#)]
27. Zheng, J.; Ma, L.; Wu, Y.; Ye, L.; Shen, F. Nonlinear Dynamic Soft Sensor Development with a Supervised Hybrid Cnn-Lstm Network for Industrial Processes. *ACS Omega* **2022**, *7*, 16653–16664. [[CrossRef](#)] [[PubMed](#)]
28. Gu, X.; Wang, S. Bayesian Takagi–Sugeno–Kang Fuzzy Model and Its Joint Learning of Structure Identification and Parameter Estimation. *IEEE Trans Ind. Inform.* **2018**, *14*, 5327–5337. [[CrossRef](#)]
29. Moya Almeida, V.; Diezma Iglesias, B.; Correa Hernando, E.C. Artificial Neural Networks and Gompertz Functions for Modelling and Prediction of Solvents Produced by the *S. Cerevisiae* Safale S04 Yeast. *Fermentation* **2021**, *7*, 217. [[CrossRef](#)]
30. Florea, A.; Sipos, A.; Stoisor, M.-C. Applying Ai Tools for Modeling, Predicting and Managing the White Wine Fermentation Process. *Fermentation* **2022**, *8*, 137. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.