Inferring Galactic Parameters from Chemical Abundances: A Multi-star Approach

Oliver H. E. Philcox1,2 and Jan Rybizki3

1 Department of Astrophysical Sciences, Princeton University, Princeton, NJ 08544, USA; ohep2@alumni.cam.ac.uk
2 Center for Astrophysics | Harvard & Smithsonian, 60 Garden Street, Cambridge, MA 02138, USA
3 Max Planck Institute for Astronomy, Königstuhl 17, D-69117 Heidelberg, Germany

Received 2019 September 2; revised 2019 October 22; accepted 2019 October 25; published 2019 December 5

Abstract

Constraining parameters such as the initial mass function high-mass slope and the frequency of Type Ia supernovae is of critical importance in the ongoing quest to understand galactic physics and create realistic hydrodynamical simulations. In this paper, we demonstrate a method for precisely determining these using individual chemical abundances from a large set of stars, coupled with some estimate of their ages. Inference is performed via the simple chemical evolution model Chempy in a Bayesian framework, marginalizing over each star’s specific interstellar medium parameters, including an element-specific “model error” parameter to account for inadequacies in our model. Hamiltonian Monte Carlo methods are used to sample the posterior function, which is made possible by replacing Chempy with a trained neural network at negligible error. The approach is tested using data from both Chempy and the IllustrisTNG simulation, showing subpercent agreement between inferred and true parameters using data from up to 1600 individual stellar abundances. For IllustrisTNG, the strongest constraints are obtained from metal ratios, competitive with those from other methods including star counts. Analysis using a different set of nucleosynthetic yields shows that incorrectly assumed yield models can give non-negligible bias in the derived parameters; this is reduced by our model errors, which further show how well the yield tables match the data. We also find a significant bias from analyzing only a small set of stars, as is often done in current analyses. The method can be easily applied to observational data, giving tight bounds on key galactic parameters from chemical abundances alone.

Unified Astronomy Thesaurus concepts: Astrochemistry (75); Interstellar abundances (832); Chemical enrichment (225); Chemical abundances (224); Milky Way Galaxy physics (1056); Astrostatistics (1882); Astrostatistics tools (1887); Stellar abundances (1577); Galactic abundances (2002)

1. Introduction

The construction of steadily more accurate large-scale galactic and cosmological simulations is an ongoing effort in the astronomical community (e.g., Few et al. 2012; Grand et al. 2017; Pillepich et al. 2018a), yet all these simulations rest upon potentially unjustified assumptions about the values of galactic parameters, which control a number of effects, including the birth and death rates for various types of stars. Two crucial unknowns are the shape of the initial mass function (IMF), setting the mass distribution of stars born from the interstellar medium (ISM), and the rate of Type Ia supernovae (SN Ia) explosions.

Despite a wealth of work on the subject, the constraints on these parameters remain weak, although it is clear that their values play an important part in determining chemical evolution tracks (Romano et al. 2005; Mollá et al. 2015; Vincenzo et al. 2015). For example, a large variety of high-mass IMF slopes have been postulated (Côté et al. 2016, Table 7), with a steeper-than-canonical gradient being suggested by a range of studies using varied data sets, including M31 star counts (Weisz et al. 2015), galactic disk structure (Rybizki & Just 2015), and analysis of thin-disk stars (Chabrier et al. 2014). In addition, the IMF slope may itself be a function of metallicity, introducing further complexity (e.g., Gutcke & Springel 2019; Martín-Navarro et al. 2019). There is also contention regarding the choice of SN Ia delay-time distribution and normalization (Maoz et al. 2010, 2012; Jiménez et al. 2015), which plays a crucial role in the enrichment of the ISM.

Given the growing wealth of stellar observational abundance data (e.g., from APOGEE; Majewski et al. 2016), this would seem to be a key data set with which to constrain galactic parameters, and previous work has contributed to this, using either chemical abundances from a small set of stars (Rybizki et al. 2017a, hereafter R17) or entire chemical evolution tracks (Mollá et al. 2015; Rybizki 2018), although only through use of binned statistics. Many of these analyses are unable to implement a fully Bayesian approach, which has the advantage of giving numerical constraints with the ability to marginalize out nuisance parameters. Thanks to the relatively tight bounds that can now be placed on stellar ages (Feuillet et al. 2016; Martig et al. 2016; Ness et al. 2016), we may begin to explore the huge expanses of data provided by the individual chemical abundances of a large set of stars, which can be used to place strong constraints on galactic parameters.

The principal goal of this work is to demonstrate how we may use modern statistical techniques and machine learning in tandem with a simple galactic chemical evolution (GCE) model in a Bayesian framework to infer global galactic parameters from a set of stars. We focus on two key parameters: the high-mass slope of the Chabrier (2003, Table 1) IMF and the rate of SN Ia explosions per unit mass, both of which we assume to be constant across the galaxy. Our primary framework is based on the Chempy model, a simple GCE parameterization that is able to predict stellar chemical abundances given a number of galactic parameters. Previous work with Chempy (R17; Feuillet et al. 2018; Philcox et al. 2018; hereafter P18) has concentrated on its application to protosolar abundances; here we aim to extend this by using multiple stellar data points. The larger volume of data should be able to give tighter statistical constraints on those parameters that are held fixed across the
galaxy, but complexity is added because we must allow each star to carry its own set of local ISM parameters.

Our inference makes use of the modern statistical technique Hamiltonian Monte Carlo (HMC; Neal 2012) sampling, made possible by the replacement of the Chempy function with a trained neural network following P18. For high-dimensional posterior functions, HMC gives much faster sampling than conventional Markov chain Monte Carlo (MCMC) methods, with the neural network allowing for analytic differentiability. We test our analysis using mock observations drawn first from Chempy, then from large-scale hydrodynamical simulations to ensure that we recover the correct parameters even for models with a completely different treatment of ISM physics. The methods could naturally be extended to any fast and flexible GCE model, not just Chempy. The code used in this paper builds upon the ChempyScoring module (P18) and has been made publicly available as a new package, ChempyMulti (Philcox & Rybizki 2019), including a comprehensive tutorial covering both the Chempy model and HMC inference.

We begin by describing the GCE models in Section 2 before considering how to use machine learning to optimize the latter in Section 3. Sections 4 and 5 discuss the Bayesian statistics and our methods for sampling from them, before we present the results for two sets of mock data in Section 6. We conclude with a summary in Section 7. In the appendix, we give technical details of the neural network, a general overview of HMC sampling, and representative sampling plots in Appendices A–C, respectively.

2. GCE Models

In order to infer GCE parameters, we need a simple physical model that takes these as inputs and can be inserted into a Bayesian framework. In addition, if we are interested in testing the validity of our approach, we require a high-resolution simulation that (a) has outputs that may be used in place of observational data (in the form of stellar ages and protostellar abundances) and (b) has well-defined values of the global parameters that we can compare to the inferred parameters. Galactic-scale hydrodynamical simulations can be effectively used in this context. We thus need two independent GCE models in our analysis, of high and low complexity, respectively.

2.1. IllustrisTNG

In this paper, we use mock observational data derived from the IllustrisTNG (hereafter TNG) magnetohydrodynamical simulations (Marinacci et al. 2018; Naiman et al. 2018; Nelson et al. 2018, 2019; Pillepich et al. 2018a; Springel et al. 2018). These are a successor to the Illustris simulations (Vogelsberger et al. 2014; Nelson et al. 2015), using an updated physical and chemical model, including new galactic physics and an improved set of nucleosynthetic yields. Here, we are principally interested in the TNG100-1 simulation (of dimension L ∼ 110 Mpc), which provides the highest resolution for publicly available data at a baryonic mass resolution of 1.4 × 10^9 M☉ (Nelson et al. 2019). Importantly, both the high-mass slope of the Chabrier (2003, Table 1) IMF and the SN Ia normalization (equal to the number of SN Ia formed in 13.8 Gyr per unit mass) are fixed parameters in TNG, with values α_{IMF} = -2.3 and N_{Ia} = 1.3 × 10^{-3} M_{☉}^{-1}, respectively (Pillepich et al. 2018b).

The simulation consists of a vast amount of galaxies (clustered in dark matter halos), each of which hosts a large number of subparticles, which can be considered as different stellar environments, subject to some set of latent parameters describing the interstellar medium (ISM) therein. For each subparticle, TNG records the typical birth time of a star in this location, as well as its initial abundances. This thus provides an excellent set of mock stellar abundance data. This is similar to the set found in a typical observational data set such as the APOGEE catalog (Majewski et al. 2016), but no post-birth abundance corrections are required. These data, coupled with the fixed galactic parameters, allow us to test the validity of our full analysis pipeline including the approximations made by our simple GCE model used for Bayesian inference.

2.2. Chempy

Chempy (R17) is a simple one-zone GCE model that computes the chemical evolution of a region of the ISM throughout cosmic time. Through use of published nucleosynthetic yield tables for three key processes (SN Ia and SN II explosions, and AGB stellar feedback) and a small number of parameters controlling simple stellar populations (SSPs) and ISM physics, the model predicts ISM chemical element abundances at time T, which can be matched to protostellar abundances for a star born at the same time T that act as a proxy for the ISM abundances. Despite its simplicity, the model has been shown to work well in a variety of contexts (e.g., Feuillet et al. 2018), especially due to its speed. As discussed below, this speed is greatly boosted by use of machine learning, first demonstrated in P18.

Here, we allow six Chempy parameters to vary freely, as shown in Table 1. These may be split into three groups:

1. A: Global galactic parameters. These describe SSP physics, and comprise the high-mass Chabrier (2003) IMF slope, α_{IMF}, and (logarithmic) Type Ia supernovae normalization, log_{10}(N_{Ia}). We assume these to be constant across both the variety of ISM environments found in a typical galaxy and cosmic time, thus they are treated as star-independent in this analysis. (While log_{10}(N_{Ia}) is constant with respect to time by definition, it being simply a normalization constant, there is some evidence for α_{IMF} varying as a function of time or metallicity (Chabrier et al. 2014; Clauwens et al. 2016; Gutcke & Springel 2019; Martín-Navarro et al. 2019), although this is not included in the TNG model.) We adopt the same broad priors as P18 for these variables (as stated in Table 1), noting that they fully encompass the values chosen by the TNG simulation.

2. Θ_i: Local galactic parameters. These describe the local physics of the ISM and are hence specific to each stellar environment, indexed by i. As defined in R17, these include the star formation efficiency (SFE) parameter, log_{10}(SFE), log_{10}(SFR peak), which controls the peak of the star formation rate (SFR), and the outflow feedback fraction, x_{out} (controlling the fraction of stellar outflow that is fed to the simulation gas reservoir; the remainder enriches the local ISM). We adopt broad priors for all parameters, and as in P18, do not allow the SN Ia delay-time distribution to vary freely, fixing it to the TNG form.
### Table 1

Free *Chempy* Parameters for Each Star, with Their Prior Values and Gaussian Widths

| Parameter                  | Description                                                                 | $\bar{\eta}_{\text{prior}} \pm \eta_{\text{prior}}$ | Limits                        | Approximated prior based upon: |
|----------------------------|-----------------------------------------------------------------------------|----------------------------------------------------|-------------------------------|--------------------------------|
| $\alpha_{\text{IMF}}$      | High-mass slope of the Chabrier (2003) IMF                                  | $-2.3 \pm 0.3$                                    | $[-4, -1]$                    | Chabrier (2003, Table 1)       |
| $\log_{10}(N_{\text{Ia}})$ | Number of SN Ia exploding per $M_{\odot}$ over 15 Gyr                      | $-2.75 \pm 0.3$                                   | $[-5, -1]$                    | Maoz & Mannucci (2012, Table 1) |

### Table 2

Nucleosynthetic Yield Tables Used in this Analysis, Matching those of the TNG Simulation (Pillepich et al. 2018b, Table 2)

| Type       | Yield Table         |
|------------|---------------------|
| SN Ia      | Nomoto et al. (1997) |
| SN II      | Kobayashi et al. (2006), Portinari et al. (1998) |
| AGB        | Karakas (2010), Doherty et al. (2014), Fishlock et al. (2014) |

3. $\{T_i\}$: Stellar birth times. This is the time in Gyr at which a given star is formed from the ISM, and we assume that its protostellar abundances match the local ISM abundances at $T_i$. Unlike in previous *Chempy* analyses, this is required to be a free parameter (because it is rarely known to high precision), and we adopt individual priors from mock observational data for each star. For real data sets, we can use the computed age estimates to define this, e.g., Ness et al. 2016. The *Chempy* code has been adapted to take this as an input, allowing the simulation to stop and return abundances at $T_i$.

The separability of local (ISM) parameters and global (SSP) parameters is motivated by recent observational evidence: Ness et al. (2019) find that the elemental abundances of red clump stars belonging to the thin disk can be predicted almost perfectly from their age and [Fe/H] abundance. This implies that the key chemical evolution parameters affecting the elemental abundances (SSP parameters and yield tables) are held fixed, while ISM parameters vary smoothly over the thin disk (which offsets the metallicity for different galactocentric radii). Similarly, Weinberg et al. (2019) find that ISM parameter variations are deprojected in the [X/Mg] versus [Mg/H] plane (their Figure 17) and that abundance tracks in that space are independent of the stellar sample’s spatial position within the Galaxy (their Figure 3).

To avoid unrealistic star formation histories (which are very “bursty” for early stars), we additionally require that the SFR (parametrized by a $\Gamma$ distribution with shape parameter $a = 2$) at the maximum possible stellar birth time (13.8 Gyr) should be at least 5% of the mean SFR, ensuring that there is still a reasonable chance of forming a star at this time step. This corresponds to the constraint $\log_{10}(\text{SFR}_{\text{peak}}) > 0.294$. For this reason, a truncated normal prior will be used for the SFR parameter. Furthermore, we constrain $T_i$ to the interval [1, 13.8] Gyr (assuming a universe age of 13.8 Gyr as in the TNG cosmology), ignoring any stars that formed before 1 Gyr, which is justified as these are expected to be rare.

To ensure maximum compatibility with TNG, we adopt their nucleosynthetic yield tables in *Chempy*, for enrichment by SN Ia, SN II, and AGB stars. The yields we used are summarized in Table 2 and match those of Pillepich et al. (2018b, Table 2), and we note that the SN II yields are renormalized such that the IMF-weighted yield ratios at each metallicity are equal to those from the Kobayashi et al. (2006) mass-range models alone. *Chempy* uses only net yields, such that they provide only newly synthesized material, with the remainder coming from the initial SSP composition. These tables may not represent true stellar chemistry well, and the effects of these are examined in Section 6.2 by performing inference using an alternative set of yields. For the analysis of observational data, we would wish to use the most up-to-date yields, such as the Karakas & Lugano (2016) AGB yields, and carefully chose elements that are known to be well reproduced by our current models (e.g., shown by Griffith et al. 2019; Weinberg et al. 2019), although this is not appropriate in our context. To facilitate the best comparison with TNG, we further set the maximum SN II mass as 100 $M_{\odot}$ (matching the IMF upper mass limit), adopt stellar lifetimes from Portinari et al. (1998), and do not allow for any “hypernovae” (in contrast to P18).

TNG only tracks nine elements in their analysis: C, Fe, H, He, Mg, N, Ne, O, and Si, reporting the mass fractions of each (Pillepich et al. 2018b). In our analysis we principally compare the logarithmic abundances $[X/Fe]$ and $[Fe/H]$ (defined by $[X/Y] = \log_{10}(N_X/N_Y)_{\odot} - \log_{10}(N_X/N_Y)_{\odot}$) for the number fraction $N_X$ of element X, where $\odot$ denotes the solar number fractions of Asplund et al. (2009). This uses H for normalization, thus we are left with $n_{\text{var}} = 8$ independent elements that must be tracked by *Chempy*.\(^7\) In this paper,}\(^6\) In analyses using a set of old stars, for example, this restriction is not appropriate because it forces there to still be a non-negligible SFR today. In these cases, the condition should be relaxed.
Chempy is used as the principal GCE model. With the modifications described above, this allows for fast prediction of TNG-like chemical abundances for a given set of galactic parameters. It is important to note that the two GCE models have very different parameterizations of galactic physics, with TNG including vastly more effects, thus it is not certain a priori how useful Chempy will be in emulating the TNG simulation, although its utility was partially demonstrated in P18. This is a necessary test to prepare for an inference on real data.

3. Neural Networks

Despite the simplifications made by emulating the TNG simulations with the simple GCE model Chempy, we will still have difficulties sampling the distribution of the global parameters \( \mathbf{\Lambda} = \{ \alpha_\text{tMF}, \log_{10}(N_{\text{H}_2}) \} \) due to the run-time of Chempy and the high-dimensionality of the parameter space. To ameliorate this, we use neural networks, which are fast nonlinear functions containing a large number of trainable parameters.

According to the universal approximation theorem (Csáji 2001), an arbitrarily complex smooth function can be approximated to any given level of precision by a feed-forward neural network with a finite number of neurons \( (n_{\text{neuron}}) \) and a single hidden layer that practically acts as a nonlinear interpolator. This implies that, given sufficient training data, a neural network can represent the Chempy function arbitrarily well. In essence, instead of computing the full model for each input parameter set, we pass the parameters to the network, which predicts the output abundances to high accuracy. This has two benefits:

1. Speed: The run-time of the Chempy function is \( \sim 1 \) s per input parameter set, which leads to very slow posterior sampling. With the neural network, this reduces to \( \sim 5 \times 10^{-5} \) s, and is trivially parallelizable, unlike Chempy.

2. Differentiability: The neural network has a simple closed-form analytic structure (described in Appendix A), unlike the complex Chempy model. This allows it to be differentiated, so we can sample via advanced methods (see Section 5).

Despite the additional complexity introduced by using multiple stellar data points, our network simply needs to predict the birth-time abundances for a single star (with index \( i \)) given a set of six parameters, \( \{ \mathbf{\Lambda}, \mathbf{\Theta}_i, T_i \} \). The same network can be used for all \( n_{\text{stars}} \) stars (and run in parallel), reducing a set of \( n_{\text{stars}} \) runs of Chempy to a single matrix computation (with input and output matrices being formed of the stacked parameter and abundance vectors). In this implementation (which differs from that of P18), we use a sparsely connected single-layer network with \( n_{\text{neuron}} = 40 \) neurons for each of \( n_\text{el} = 8 \) abundance outputs. This is trained with a sample of \( 10^6 \) sets of input parameters and output abundances, with hyperparameter optimization and testing performed with an independent sample of consisting of \( 5 \times 10^4 \) parameter sets. With the above choices, the network predicts abundances with an average error of \( 0.005^{+0.009}_{-0.003} \) dex, which is far below typical observational errors and even smaller away from the extreemes of parameter space. Technical details of the network and implementation are discussed in Appendix A.

4. The Statistical Model

We here extend the Bayesian model introduced in R17 to include multiple stellar data points. Consider a given star with index \( i \), born in some region of the ISM. This will carry its own set of parameters \( \{ \mathbf{\Lambda}, \mathbf{\Theta}_i, T_i \} \), where \( \mathbf{\Lambda} \) are taken to be global (hence independent of the stellar label \( i \)), but the ISM parameters \( \mathbf{\Theta}_i \) and the birth time \( T_i \) are specific to the star. Using the Chempy function (or the trained neural network), we can compute the output \( n_\text{el} \) chemical abundances \( \{ X_i^j \} \) for the \( i \)th star as

\[
\{ X_i^j \} = \text{Chempy}(\mathbf{\Lambda}, \mathbf{\Theta}_i, T_i),
\]

where \( j \) indexes the chemical element. These can be compared against observations, with measured abundances \( d_i^j \) and corresponding Gaussian errors \( \sigma_{i,\text{obs}}^j \), jointly denoted \( D_i = \{ d_i^j, \sigma_{i,\text{obs}}^j \} \). In addition, we add a star-independent model error parameter \( \sigma_{\text{model}}^j \) for each element, which accounts for imperfections in our GCE model (e.g., due to incorrect yields) and is allowed to vary freely.\(^8\) This allows the inference to give less weight to elements that are empirically found to fit the data less well. The \( i \)th star likelihood is thus simply a product over \( n_\text{el} \) Gaussians,

\[
\mathcal{L}(D_i | \mathbf{\Lambda}, \mathbf{\Theta}_i, T_i, \Sigma) = \prod_{j=1}^{n_\text{el}} \frac{1}{\sqrt{2\pi(\sigma^j_{i,\text{tot}})^2}} \times \exp \left( -\frac{(d_i^j - X_i^j)^2}{2(\sigma^j_{i,\text{tot}})^2} \right),
\]

where \( \sigma^j_{i,\text{tot}} = \sqrt{(\sigma^j_{i,\text{obs}})^2 + (\sigma^j_{\text{model}})^2} \), combining errors in quadrature and denoting the model errors by \( \Sigma = \{ \sigma^j_{\text{model}} \} \).

For a collection of \( n_{\text{stars}} \) stellar data points with the local parameter set \( \{ \mathbf{\Theta}_i \} \) and birth times \( \{ T_i \} \), the joint likelihood is simply a product over the individual likelihoods:

\[
\mathcal{L}(D_1, \ldots, D_{n_{\text{stars}}}) = \prod_{i=1}^{n_{\text{stars}}} \mathcal{L}(D_i | \mathbf{\Lambda}, \mathbf{\Theta}_i, T_i, \Sigma).
\]

The full posterior function is derived simply via Bayes rule as

\[
\mathbb{P}(\mathbf{\Lambda}, \{ \mathbf{\Theta}_i \}, \{ T_i \}, \Sigma | \{ D_i \}) \propto \prod_{i=1}^{n_{\text{stars}}} \mathbb{P}_D(D_i | \mathbf{\Lambda}, \mathbf{\Theta}_i, T_i) \times \mathbb{P}_\mathbf{\Theta}(\mathbf{\Theta}_i) \times \mathbb{P}_T(T_i) \times \mathbb{P}_\mathbf{\Lambda}(\mathbf{\Lambda}) \times \prod_{j=1}^{n_\text{el}} p_{\Sigma}^j(\sigma^j_{\text{model}}) \times \mathcal{L}(D_1, \ldots, D_{n_{\text{stars}}}),
\]

where \( p_{\mathbb{V}}(V) \) is the prior on the variable \( V \) (belonging to the set \( \mathbb{V} \)). The priors are chosen to have the following form:

1. \( \mathbf{\Lambda} \): Gaussian priors for \( \alpha_{\text{tMF}} \) and \( \log_{10}(N_{\text{H}_2}) \) with parameters defined in Table 1.

2. \( \mathbf{\Theta}_i \): Gaussian priors for \( \log_{10}(\text{SFE}) \) and \( x_{\text{out}} \) according to Table 1 with a truncated Gaussian prior for the peak SFR parameter, restricted to \( \log_{10}(\text{SFR}_{\text{peak}}) > 0.294 \) (see Section 2.2). Although \( \mathbf{\Theta}_i \) is different for each star, each

\(^8\) This is similar to the model error introduced in P18, but we now allow it to vary between elements.
vector is taken to be a draw from a star-independent prior.\(^9\)

3. \(T_i\): Gaussian prior for each star independently. The prior parameters are set from an estimate of the stellar birth time and its variance, representing our best knowledge of this parameter. In experimental contexts, this would be found from age models (e.g., in the Cannon model (Ness et al. 2016) for red giant stars in the APOGEE (Majewski et al. 2016) survey).

4. \(\Sigma = \{\sigma_{i,model}\}: \) Half-Cauchy prior with shape parameter (standard deviation) \(\beta_{model} = 0.01\). This choice of prior (defined for \(\sigma_{i,model} \geq 0\)) allows for arbitrarily small errors, as well as those much greater than the observational errors (\(\sim 0.05\) dex) for poorly reproduced elements.

In statistical language, the model can be expressed as

\[
\begin{align*}
\Lambda &\sim p_\Lambda = N(\mu_\Lambda, \sigma_\Lambda) \\
\Theta_i &\sim p_\Theta = N^*(\mu_\Theta, \sigma_\Theta) \\
T_i &\sim p_{T_i} = N(\mu_{T_i}, \sigma_{T_i}) \\
\sigma_{i,model} &\sim p_{\Sigma} = \text{Half - Cauchy}(\beta_{model}) \\
\{X_i^j\} &\sim \text{Chempy}(\Lambda, \Theta_i, T_i) \\
\sigma_{i,tot} &\sim \sqrt{(\sigma_{i,obs}^2 + (\sigma_{i,model}^2)^2) \\
X_i^j &\sim N(d_i^j, \sigma_{i,tot}^j),
\end{align*}
\]

where \(N^*\) indicates a possibly truncated Gaussian (for the SFR parameter). In total, we have \(2 + 4n_{stars} + n_{el}\) free parameters to be inferred from \(n_{el}\) data points, given \(6 + n_{stars}\) individual prior distributions. This is summarized in Figure 1, in the form of a probabilistic graphical model (PGM), which shows the relationship between all variables and hyperparameters.

5. Sampling Techniques

To determine the optimal values of the global galactic parameters (\(\Lambda\)), we must sample the posterior of Equation (5). In previous work (R17; P18), this was achieved using ensemble sampling MCMC using the emcee package (Foreman-Mackey et al. 2013). The authors of emcee note that this is not appropriate for sampling high-dimensional parameter spaces, thus here, where the dimensionality scales with \(n_{stars}\), we must find an alternative sampler. Gibbs sampling (Geman & Geman 1984) is one option, where marginal posterior functions are used to iteratively first update the global \(\Lambda\) and \(\Sigma\) parameters and then the local \(\{\Theta_i, T_i\}\) parameters, based on a Metropolis–Hastings sampling approach (Hastings 1970). However, this is difficult to use in practice, due to (a) the requirement of knowing the marginal posterior functions (e.g., \(P(\Lambda|\{\Theta_i, T_i\}), \Sigma, \{D_i\})\), (b) the large number of tunable parameters, and (c) slow convergence.

Here we principally consider the modern sampling technique Hamiltonian Monte Carlo (HMC; Neal 2012), which uses posterior function gradients to sample much more efficiently than canonical MCMC methods. This can also sample much higher-dimensional posteriors than ensemble sampling. The basic premise (explained in more detail in Appendix B) is as follows. In standard MCMC approaches, given a current position in the MCMC chain, the next position is chosen via a random jump such that the chain traverses a random walk in parameter space. By introducing additional momentum parameters, we can choose samples in a more efficient manner similar to a rocket exploring the space around a planet by traversing orbits of constant energy, then making random jumps in energy rather than just jumping between positions at random. This requires the posterior function to be differentiable, however, which is seldom possible for complex astronomical models. In this context, the replacement of Chempy by a trained neural network gives a trivially differentiable model because the network is a simple function of matrices and tanh functions, thus HMC can be used in our context.

In practice, this is implemented using the Python PyMC3 package (Salvatier et al. 2016),\(^{10}\) using the automatic differentiation routines from theano (Al-Rfou et al. 2016) to compute the posterior gradients. HMC sampling is performed via the no U-turn sampler (NUTS; Hoffman & Gelman 2011)

---

\(^9\) A more refined approach would be to assume a full hierarchical structure, where each \(\Theta_i\) was a draw from some distribution whose parameters were allowed to vary freely, themselves drawn from a hyperprior, e.g., promoting the mean and variance of \(p_{\theta}\) to be free parameters. This adds additional complexity and is not explored in this paper.

\(^{10}\) docs.pymc.io/
using 16,000 chain samples with a desired sample-acceptance rate of 0.9. The sampler uses $2 \times 10^4$ initialization steps (which set the start point of the Markov chain) and 2000 tuning steps (to adjust internal parameters and stabilize the Markov chain), with sampling expedited by running multiple smaller chains in parallel on different CPUs, which can then be combined.

In the case of very large $n_{\text{stars}}$, the dimensionality of our problem becomes large, and we find that even HMC requires an unwieldy sampling time. For this reason, we restrict to $n_{\text{stars}} \leq 200$ for the HMC analysis to ensure sampling can be done in a few tens of CPU hours. For a larger sample of stars, we may look to approximate sampling methods, such as automatic differentiation variational inference (ADVI; Kingma & Welling 2013; Kucukelbir et al. 2016; Roeder et al. 2017). This is briefly discussed in Appendix B, and a simple form (mean field ADVI) is used for the NUTS initialization steps. We found that $n_{\text{stars}} = 200$ gives well-constrained posterior parameter estimates in this paper, thus we do not implement ADVI here.

### 6. Results

In this section we apply the statistical techniques described in Section 5 to the posteriors of Section 4 to infer the global galactic parameters $\alpha_{\text{IMF}}$ and $\log_{10}(N_{\text{th}})$. To demonstrate the utility of our method, we compare the derived global parameters with the true values, using three mock data sets:

1. A data set created by 
   Chempy with the same nucleosynthetic yield tables as for the neural network training. This is used to test the sampling techniques and neural networks.
2. A data set created by 
   Chempy with different yield tables to that of the neural network. This is used to test the dependence of our inference on the yield tables.
3. A data set derived from stellar particles taken from a galaxy in the TNG simulation (yields are the same as in case 1). This is used to test the dependence of our inference on the galactic physics parameterization.

In each case we obtain a set of stellar birth times and chemical abundances, which, to fully represent observational data, must be augmented with errors. In line with typical APOGEE (Majewski et al. 2016) abundance data, we conservatively assume a uniform Gaussian error of 0.05 dex in the [Fe/H] and [X/Fe] values. In addition, we assign a 20% fractional error to each birth-time measurement ($T_i$), roughly matching that obtained in current analyses using APOGEE data (Ness et al. 2016). Mock observed abundances and birth-times are drawn from Gaussian distributions about their true values with the above errors, and we disregard any stars with mock observed birth times (i.e., the prior means) $\mu_{T_i} \notin [1, 13.8]$ Gyr. The outcome of this mock data creation is a set of 200 mock stars, all with relevant observational abundances and birth times, emulating a real data set. These data sets have been made freely available online alongside a tutorial showing their format and usage.

#### 6.1. Mock Data from Chempy

To create the Chempy mock data, we first set the values of the global galactic parameters as $\alpha_{\text{IMF}} = -2.3$ and $\log_{10}(N_{\text{th}}) = -2.89$, matching those used by TNG (Pillepich et al. 2018b). Using the priors in Table 1, we then create a set of 200 random draws of the local parameters $\Theta_i = \{\log_{10}(\text{SFE}), \log_{10}(\text{SFR peak}), x_{\text{out}}\}$, additionally drawing $T_i$ uniformly from the range [2, 12.8] Gyr, to minimize overlap with the neural network training birth-time limits when observational uncertainties are included. Each set of parameters is passed to the Chempy function, producing eight output true chemical element abundances that are then augmented with errors, as above.

Following this, the methods of Section 5 are used to infer the posterior distribution of $\Lambda$ by sampling the full high-dimensional parameter space via the HMC algorithm. Here, Chempy is being used both to create and fit the data, thus there is no mismatch between observations and sampler in terms of physics parameterization or yield tables. This should imply small model errors (i.e., $\sigma_{\text{model}} \rightarrow 0$), although the model errors are retained in the inference as a useful test. Analysis is performed for a selection of $n_{\text{stars}} \in [1, 200]$. To illustrate the bias created by using only a small selection of stars, we split a sample of 200 stars into non-intersecting subsamples of size $n_{\text{stars}}$ and perform the inference separately on each (i.e., we perform 100 one-star analyses, 50 two-star analyses, etc.). In our implementation (using parallel sampling across 16 cores), the analysis of each subsample has a run-time ranging from ~1 CPU minute (for $n_{\text{stars}} = 1$) to ~40 CPU hours (for $n_{\text{stars}} = 200$) on a modern machine.

The resulting posterior distribution parameters of $\Lambda$ are summarized in Figure 2(a) and Table 3(a). For the measurement of global parameters in a subsample of stars we note two contributions to the variance: (a) the intrinsic statistical variance from the width of the posterior distribution for $\Lambda$ (shown by the shaded regions in the plot), and (b) the sample variance arising from the bias caused by analyzing only a small set of stars (shown by the spread of individual posterior medians in the plot). For small $n_{\text{stars}}$, the effects have similar magnitude, with sample variance contributing ~4% to the total uncertainty of each realization for $n_{\text{stars}} = 1$ (quantified by the standard deviation of the median posterior parameter estimates between subsamples). For large subsamples, where we include stars from a large variety of ISM environments, the effect is subdominant, however. This implies that measuring galactic parameters from a single star can give significantly biased results, which is important to take into account when considering single-star analyses such as R17.

Considering the average over all subsamples at fixed $n_{\text{stars}}$ (as in Table 3(a)), the median of the posterior inferences are seen to be in full agreement with the true values in all cases, given the statistical errors. For $n_{\text{stars}} \geq 5$, this is additionally true for the estimates from individual subsamples, confirming that the sample variance effect is of only minor importance at large $n_{\text{stars}}$. As expected, the statistical widths of the posterior distributions shrink as $n_{\text{stars}}$ increases because the number of individual data points (here $n_{\text{data}} = 8n_{\text{stars}}$) becomes large compared to the number of free parameters ($2 + n_{\text{el}} + 4n_{\text{stars}} = 10 + 4n_{\text{stars}}$). For $n_{\text{stars}} = 200$, we obtain bounds of $\alpha_{\text{IMF}} = -2.31 \pm 0.01$, $\log_{10}(N_{\text{th}}) = -2.90 \pm 0.01$, which is fully consistent, as before.\footnote{We note that the choice of stellar age distribution is unimportant here as long as all birth times are inside the neural network training limits.}

\footnote{Because we only use a single subsample for the $n_{\text{stars}} = 200$ analysis, the sample variance cannot be determined. Given the general trend with $n_{\text{stars}}$, however, we expect it to be small.}

\footnote{11 github.com/oliverphilcox/ChempyMulti}

---

**References**

1. Majewski et al. 2016
2. Roeder et al. 2017
3. Kucukelbir et al. 2016
4. Chempy et al. 2016
5. APOGEE (Majewski et al. 2016)
6. Ness et al. 2016
7. Majewski et al. 2016
8. Fe/Fe = 0.05 dex
9. $\mu_{T_i} \notin [1, 13.8]$ Gyr
10. $\mu_{T_i} \notin [1, 13.8]$ Gyr
11. github.com/oliverphilcox/ChempyMulti
Analysis of the posterior model errors, \( \Sigma \equiv \{ \sigma^j_{\text{model}} \} \), is performed in Figure 3(a), showing the full posterior distributions for \( n_{\text{stars}} = 200 \), and Table 4(a), summarizing the inferred parameters for a range of data-set sizes. We first note the model errors to be approximately independent of the element label \( j \), as predicted. (We expect all elements to be equally reliable because there is no mismatch between data and sampling model). The distributions are clearly centered on zero, and are similar in form to the priors (half-Cauchy distributions with standard deviation \( \beta = 0.01 \) dex), although they become sharper as \( n_{\text{stars}} \) increases. Taking the median across all elements and subsamples at fixed \( n_{\text{stars}} \), the average standard deviation of \( \sigma^j_{\text{model}} \) falls from \( \approx 0.05 \) dex at \( n_{\text{stars}} = 1 \) to \( \approx 0.005 \) dex at \( n_{\text{stars}} = 200 \), significantly below the prior value. As \( n_{\text{stars}} \) increases, so does the number of independent data points, leading to smaller statistical error and hence a reduced standard deviation (given that the prior is peaked at zero). This behavior is fully consistent with the \( \sigma^j_{\text{model}} \to 0 \) limit, with no preference shown for nonzero model errors.

We may also consider the constraints that may be placed on the stellar birth times from this analysis. The posterior estimates of \( T_i \) are highly consistent with the true values, with...
a fractional deviation of $-0.02^{+0.16}_{-0.15}$ ($0.00 \pm 0.15$) for $n_{\text{stars}} = 1$ ($n_{\text{stars}} = 200$), averaging across all 200 stars. In addition, the posterior distributions are somewhat narrower than the priors, with fractional widths of $0.16^{+0.01}_{-0.02}$ ($0.14 \pm 0.02$) for $n_{\text{stars}} = 1$ ($n_{\text{stars}} = 200$), compared to the prior width of 20%. These constraints are far weaker than those of the global parameters, showing little variation with the subsample size. This is because the birth times belong to the set of local variables (along with the three ISM parameters), which must be constrained by only $n_{\text{el}} = 8$ data points, unlike the global parameters, which are constrained by all $n_{\text{stars}}n_{\text{el}}$ abundances. For larger $n_{\text{stars}}$, each individual data point has less effect on $\Lambda$, thus the constraining power of the data on the local parameters increases slightly, although we are still limited by $n_{\text{el}}$. To obtain sharper constraints, we need only increase the number of elements that are analyzed. In applications of this method to observational data, our age analysis would be aided by models of surface chemical abundance change (e.g., Martig et al. 2016), as well as by implementation of more nucleosynthetic processes, in order to provide age-sensitive elements (Nissen 2016; Spina et al. 2018; Titterenko et al. 2019), although in the context of GCE models this usually depends on the galactic component under investigation (e.g., Kobayashi & Nakasato 2011; Nissen & Schuster 2011).

From the above, it is clear that the latter part of our analysis works as expected, with the sampler able to correctly (and precisely) infer global parameters from data that use the same physical model and yield tables, despite only placing weak constraints on the local parameters. By increasing the number of stars (or the number of chemical elements), we can obtain tighter bounds on global parameters and reduce the bias caused by the choice of subsample. At this stage, however, it is not clear whether this will extend to samples drawn from simulations (or universes) that do not obey the same evolutionary model as Chempy.

6.2. Mock Data with an Incorrect Yield Set

In the real universe, the chemical yields from stellar nucleosynthetic processes will not exactly match those tabulated in our yield tables (Table 2). To investigate the effect of this, we consider an analysis using mock data created again by Chempy, but with a different set of nucleosynthetic yields. The yield tables we used are listed in Table 5 and were chosen to ensure that contributions to all processes differ at $\mathcal{O}(10\%)$.

In Figure 4 we visualize both yield sets.
fractional mass returned to the ISM by each nucleosynthetic process over 13.8 Gyr for an SSP formed at solar metallicity. The mean deviation between the yield sets is \(\sim 20\%\), both for the total mass return and for that from the individual nucleosynthetic processes. The greatest differences are for N, with a \(\sim 60\%\) shift in the dominant (AGB) nucleosynthetic channel, although we also note large changes to the total yield for O and Si (around 40\%). There is additionally a slight increase in the Fe yield for the new yields relative to TNG, which will affect all [X/Fe] abundances via the normalization.\(^{15}\)

Using these yields, we constructed mock data using Chempy as in Section 6.1 and performed an HMC inference with the same neural network as before (which was trained with the original TNG yields). Data were thus created with the alternative yield set, but analyzed assuming TNG yields, allowing us to explore the impact of incorrectly assumed yield tables on the output parameter distributions.

The inference results are summarized in Table 2(b) and Table 3(b) in the same manner as above. Like before, the sample and statistical variances are seen to decrease as a function of \(n_{\text{stars}}\), although we note larger variances in all cases because the data are less constraining (due to mismatches between observations and model, which increase the model error and thus decrease the constraining power). Notably, for \(n_{\text{stars}} \gtrsim 50\), the posterior parameter distributions become inconsistent with the true values, with 68\% confidence intervals of \(\delta_{\text{IMF}} = -2.22 \pm 0.01\) and \(\log_{10}(N_{\text{He}}) = -2.96 \pm 0.02\) obtained for \(n_{\text{stars}} = 200\) (ignoring greatly subdominant sample variance) compared to true values of \(-2.3\) and \(-2.89\), respectively. Because the sampler assumes different chemistry to that of the data, a run of Chempy using the true values of the SSP and ISM parameters will not reproduce the observational abundances exactly, even in the absence of observational errors. Instead, it is likely that a closer match between Chempy predictions and observations will be obtained using a slightly different set of parameters, leading to a bias in the derived posterior parameters. This is partially ameliorated by the inclusion of free model errors, which have the effect of downweighting elements that fit the data less well. If these are not implemented (i.e., setting \(\delta_{\text{model}}^j = 0\) for all \(j\)), the fractional bias is significantly increased, giving \(\delta_{\text{IMF}} = -2.374 \pm 0.005\) and \(\log_{10}(N_{\text{Fe}}) = -3.11 \pm 0.01\) for \(n_{\text{stars}} = 200\), demonstrating their utility for real analyses. In addition, when the true yield set is not known, the bias may be approximated by running the inference multiple times with different yield tables to give an empirical "yield set bias" that can be combined with the sources of uncertainty discussed above.\(^{15}\)

\(^{15}\) In principle, this could be ameliorated by performing inference using the metal mass fractions themselves rather than the abundances. The advantage of our approach is that most abundances are insensitive to the metallicity of the star (except for [Fe/H] and [He/Fe]) because they depend only on metal mass ratios.
Figure 3(b) and Table 4(b) show the posterior distributions of the $\sigma^2_{\text{model}}$, as in the previous section. Unlike before, we observe a strong preference for nonzero model errors, especially for C, N, and Si abundances, which have median values significantly greater than the observational errors (0.05 dex). This indicates that our model is unable to reproduce the observed abundances of these elements. In all three cases, we have significant differences between the alternative and TNG yields in the dominant nucleosynthetic process (see Figure 4), justifying these results.\footnote{We cannot directly identify the elements with the largest model errors to those with the largest differences in Figure 4 because Chempy abundances are a function of the yields across all metallicities and times, while the figure shows the output of a single SSP at solar metallicity. In addition, the model errors are affected by the constraining power of individual elements on the SSP and ISM parameters; incorrectly produced elements that affect the posterior constraints more strongly will have larger model errors.} In contrast, the model errors for [He/Fe] and [Fe/H] are small, indicating that there is little change to these abundances caused by changing yield set, again consistent with Figure 4 (also noting that even at late times, most of the H and He comes from the primordial gas). From the table, we note that the fractional widths of the posterior distributions shrink as $n_{\text{stars}}$ increases, while the median values increase for small $n_{\text{stars}}$ and then become independent of the subsample size. For small subsamples, it is tempting to think that the model errors will be large because there will be stars whose abundances cannot be well reproduced by the model. However, in this limit, we have a large number of free parameters to constrain with very little data, so any such errors can easily be absorbed into an ISM or SSP parameter, and the distributions will tend to reproduce the priors. As the number of data points becomes large, the data set becomes far more constraining, and we can effectively distinguish between SSP, ISM, and model error effects, causing the model error distributions to settle about their preferred values.

This analysis shows that to avoid bias in the inference of the galactic IMF and SN Ia parameters, we require yield sets that accurately represent galactic chemistry. Introduction of the model error parameters helps with this because it allows the sampler to place greater weight on those elements that are better reproduced, reducing the bias to $\sim3\%$ in this instance, despite significant differences between yield tables. Further assistance is provided by making informed choices about the yield tables, e.g., using those that best recover observational...
data sets such as the protosolar abundances (P18), and restricting to elements that are known to be well fit by current models (Griffith et al. 2019; Weinberg et al. 2019). In observational contexts, we would additionally exclude elements such as C and N, which are known to undergo significant changes in their abundance during stellar evolution (Gratton et al. 2000; Lagarde et al. 2019). A further benefit of the model errors is as a diagnostic tool; in analysis of observational data, we can assess how well individual yields match reality via the magnitude of $\sigma_{\text{model}}$ and in the (futuristic) case of highly accurate nucleosynthetic models, uncover observational biases.

6.3. Mock Data from IllustrisTNG

Figure 5 shows the chemical evolution tracks in the [Mg/Fe] vs. [Fe/H] plane for stellar particles taken from an MW-like IllustrisTNG galaxy (Pillepich et al. 2018a), colored as a function of their birth time $T_b$. This shows $\sim$40,000 individual stellar particles, with smoothed contours at $\Delta t = 1$yr shown in red. For comparison, we plot smoothed contours of the Chempy abundance distribution in black, using TNG yields and fixing the global parameters ($\alpha_{\text{MF}}$ and $\log_{10}(N_\text{IS})$) to the TNG values of $-2.3$ and $-2.89$, respectively, as in Section 6.1. Contours are created from 1000 runs of Chempy, drawing the local (ISM) parameters from the priors on $\Theta_0$ (Table 1), and the birth times, $T_b$, from the SFR model, assuming prior parameters (see Section 2.2). We caution that these are prior abundance predictions for Chempy with no fitting performed, and that each TNG stellar particle contains a range of different mass (and lifetime) stars formed at the same time and composition.

The simplified ISM physics parameterization used in Chempy does not accurately describe the physical universe. To explore the biases in the inferred galactic parameters caused by this, we apply the analysis of Section 5 to mock data drawn from the vastly more complex IllustrisTNG simulation, which was described in Section 2.1.

Here, we extract a single galaxy from the $z = 0$ snapshot of the highest resolution TNG100-1 simulation, choosing a subhalo (index 523071) with mass close to $10^{12} M_\odot$, assuming this to be similar to the Milky Way (MW). From this, we extract 200 stellar particles from the $\sim$40,000 particles that are present, each of which has a mass of $\sim 1.4 \times 10^6 M_\odot$ (Nelson et al. 2019). These act as proxies for stellar environments, giving the elemental mass fractions, $\rho_j^i / \rho^i$, and cosmological scale factor, $a_t$, at the time of stellar birth. Mass fractions are converted into $[X/Fe]$ abundance ratios using Asplund et al. (2009) solar abundances as in Chempy, with the scale factor ($a_t$) to birth time ($T_b$) conversion performed using astropy (Astropy Collaboration et al. 2013; Price-Whelan et al. 2018), assuming a $\Lambda$CDM cosmology with Planck Collaboration et al. (2016) parameters, as in TNG (Pillepich et al. 2018a). Observational errors are incorporated as above, giving a full data set that is identical in structure to the Chempy mock data.

Figure 5 shows the chemical evolution tracks in the [Mg/Fe] versus [Fe/H] plane for the full set of TNG stellar particles from the chosen galaxy. For comparison, we plot (black) contours obtained from a sample of 1000 Chempy mock data points (see Section 6.1), with birth times drawn from the range [0,13.8] Gyr, weighted by the Chempy SFR prior, each with a random realization of the local parameters, $\Theta_0$, sampled from the priors (Table 1). The abundance distributions are broadly similar between the two simulations (as expected because they use the same nucleosynthetic yields), although we note that the variance of the TNG data is much greater, especially along the [Fe/H] axis (analogous to the results of P18, Figure 7, which used a similar hydrodynamical simulation). Mismatches between the simulations are likely to result from the different ISM physics parameterizations, with TNG employing a far more realistic engine than the simple one-zone model of Chempy. A major difference is in the SFR; this is set as a one-parameter $\Gamma$-distribution in Chempy, but arises naturally from hydrodynamical processes in TNG. It is pertinent to note that the Chempy ISM parameters used in Figure 5 are chosen without knowledge of the TNG simulation; better agreement can be found using the posterior parameters for a data set, although this is costly to do for a large number of stars.

The TNG galaxy used here was deliberately chosen to have both a high-\$\alpha$ and low-\$\alpha$ chemical evolution sequence (as observed in Figure 5) to test our inference on a mock galaxy with MW-like properties. While recent simulations differ on the exact details of how bimodality develops, it is generally attributed to gas-rich mergers and different modes of star formation (Grand et al. 2018; Mackereth et al. 2018; Buck 2019; Clarke et al. 2019). In chemodynamical models, bimodality similar to the MW can also be achieved by a combination of radial migration and selection effects without the need for mergers or starbursts (Schönrich & Binney 2009; Minchev et al. 2013; Andrews et al. 2017). In the parameterization used here, Chempy can assign each star to its own ISM environment, but cannot exchange gas between environments and has no sudden star formation or infall events. We hence investigate here whether this significantly biases our inference of the SSP parameters (noting that results from Weinberg et al. (2019) justify the treatment of ISM parameters as latent variables).

The posterior distributions of $\Lambda$ obtained from HMC sampling for the TNG data set are shown in Figure 2(c) and Table 3(c). As before, the sample and statistical variances are seen to decrease as $n_{\text{stars}}$ increases, with the parameter estimates becoming statistical variance limited by $n_{\text{stars}} \approx 10$. For $n_{\text{stars}} = 1$, the statistical variance of the global parameters is similar to that found in the TNG studies of P18, which used the same chemical elements and yield tables, albeit with a different

---

17 http://www.astropy.org

18 As for the Chempy mock data, we exclude any particles with $T_b \notin [2, 12.8]$ Gyr to ensure that the true times are well separated from our training age limits, avoiding neural network errors. This removes $\sim$5% of the stars.

19 Note that we do not convolve the SFR with the stellar lifetime function to create the Chempy data for this plot. This is because we do not have individual stellar data for TNG, only the initial abundances and birth times of large stellar particles, which contain many individual stars of varied lifetimes and masses.
stellar data set, leading to a different median $\Delta$ estimate. We note a generally larger sample variance for the TNG results compared to those in previous sections; this implies that the TNG mock data set contains a broader range of stellar ISM environments than the Chempy mock data, most likely because we are not limited by the simple Chempy parameterizations. This is also demonstrated in Figure 5, where the abundance-space distribution of TNG is seen to be much broader than that of the Chempy priors. If a stellar particle outside the main Chempy realm is included in the data set by chance, the IMF slope is forced to shift to move the Chempy abundance track, leading to a greater sample variance.

For all values of $n_{\text{stars}}$, we tested, there is good agreement between the inferred parameters and their true values, obtaining best estimates of $\hat{\alpha}_{\text{IMF}} = -2.283 \pm 0.007$ and $\log_{10}(N_{\text{H}}) = -2.889 \pm 0.008$ with 200 stars, highly consistent with TNG. In addition, the posterior estimates of $\Delta$ from individual subsamples are consistent with the true values (to within $2\sigma$) for $n_{\text{stars}} \gtrsim 10$, although we caution that deviations exceeding $3\sigma$ are found when only single stars are used in the analysis. For completeness, we display the full corner plot of the 10 global parameters using $n_{\text{stars}} = 200$ in Appendix C.

To place our results in an observational context, we additionally show the constraints on $\hat{\alpha}_{\text{IMF}}$ obtained from modern analyses using star counts in M31 (Weisz et al. 2015) and the MW (Hosek et al. 2019), as well as on $\log_{10}(N_{\text{H}})$ from various observations of SN Ia (Maoz et al. 2012; Maoz & Graur 2017). While the centers of these constraints are clearly inconsistent with our results (because they use observational data, while we limit ourselves to a simulation), we may readily compare the widths of the contours to assess the constraining power of the various methods. Considering both sampling and statistical errors, our analysis gives stronger posterior constraints than the observational studies for both parameters, using $n_{\text{stars}} \gtrsim 20$. Even when we account for modeling biases (e.g., in the case of incorrect yield tables), the technique of constraining galactic parameters from individual chemical element abundances is certainly competitive.

The model errors (Figure 3(c) and Table 4(c)) exhibit similar trends with $n_{\text{stars}}$ as discussed in previous sections. In this case, however, we note small errors (below the observational error of 0.05 dex) for all abundances involving metal ratios, but large errors ($\sim 0.2$ dex) for [He/Fe] and [Fe/H] (becoming tightly constrained at large $n_{\text{stars}}$). The former shows that the metal ratios are strongly constraining (especially [N/Fe] and [Si/Fe] in this case), but the latter indicates a mismatch between TNG and Chempy either in terms of non-metal enrichment or the total metallicity (tracked by the ratio of metals to non-metals), which is consistent with the anomalous [Fe/H] behavior in Figure 5. This discrepancy will be sourced by the difference in ISM physics between the simulations; while the metal ratios are set mainly by the chemical yields, the absolute metallicity depends strongly on details such as the stellar feedback strength and star formation history, which are difficult to encapsulate within Chempy’s simple ISM physics parameterization. A likely cause of this difference is that we assume both AGB and SNe events to immediately deposit the same fraction of stellar feedback into the local ISM (i.e., $x_{\text{out}}$), which is unlikely due to the large differences in kinetic energy between the two processes. In TNG, the hotter SN feedback will be spread out far more and take more time to cool, while the colder AGB expulsions will be readily available to form new generations of stars. This will significantly affect the non-metal fractions in the simulation. One way in which to ameliorate these problems would be by introducing additional free parameters into the Chempy model, for example, including separate AGB and SNe feedback fraction parameters or controlling the size of the simulation gas reservoir. While this would likely reduce the model errors in $[\text{Fe/H}]$ and $[\text{He}/\text{Fe}]$, it would be at the expense of additional computation time, particularly if the parameters are chosen to be local, thus it has not been explored here. In our analysis, these issues are of limited importance because the large size of $[\text{Fe/H}]$ and $[\text{He}/\text{Fe}]$ model errors diminishes the impact of these abundances in the likelihood analysis. Repeating the $n_{\text{stars}} = 200$ inference without the model errors gives $\hat{\alpha}_{\text{IMF}} = -2.279 \pm 0.005$ and $\log_{10}(N_{\text{H}}) = -2.881 \pm 0.007$, showing a slight bias and $\sim 4\sigma$ tension in the IMF parameter due to the poorly reproduced $[\text{Fe/H}]$ and $[\text{He}/\text{Fe}]$ abundances.

In terms of the local parameters, the posterior distributions show a similar behavior to that of the Chempy mock data (Section 6.1). We observe a fractional error in the median inferred birth times compared to their true values of $0.00_{-0.24}^{+0.20}$ (0.01$^{+0.19}_{-0.17}$ with a fractional posterior width of 0.17$^{+0.17}_{-0.02}$ (0.16$^{+0.01}_{-0.02}$ for $n_{\text{stars}} = 1$ ($n_{\text{stars}} = 200$), only marginally narrower than the prior width of 20%. When we use only eight elements in the analysis, this technique is not capable of providing precise estimates of stellar ages (or analogously, local ISM parameters), but it is clear that we can obtain strong constraints on the global parameters when we use only weakly informative priors.

When we consider that the parameterizations of ISM physics between the two GCE models are entirely different, our inferred SSP parameters are in impressive agreement with the true values. It is pertinent to note, however, that the posterior confidence intervals on $\Delta$ are expected to shrink to zero as $n_{\text{stars}} \to \infty$ because we do not include contribution to the variances from the errors made by Chempy, thus we do expect a small bias to become apparent for very large $n_{\text{stars}}$. Due to this, extension of the method to larger $n_{\text{stars}}$ would be an interesting avenue of research. This is nontrivial, however, because the sampling time becomes long (several hours on multiple cores) for $n_{\text{stars}} \gtrsim 50$, thus we must look to alternative (approximate) sampling methods such as ADVI, allowing us to use many more data points to ensure that the error is dominated by systematics alone.

### 6.4. Potential Future Work

We briefly outline additional modifications that may need to be considered for our method to be applied to observational data. The largest obstacle arises from the uncertainties in the underlying nucleosynthetic yields, and advancement therein will improve the accuracy of the inference. This may take many forms, for instance, with the usage of empirical yields (e.g., Andrews et al. 2012; Jofré et al. 2017; Boess & Rocha-Pinto 2018; Price-Jones & Bovy 2018; Ness et al. 2019), the inclusion of the latest yield sets (e.g., Prantzos et al. 2018), the implementation of binary-star evolution effects (e.g., Abate et al. 2015; Benvenuto & Bersten 2017; Jorissen et al. 2019), or the propagation of nucleosynthetic yield uncertainties into our GCE model (Rauscher et al. 2016). Similarly, a more advanced error treatment will help to reduce bias from inevitably...
imperfect models. With some modification, our statistical analysis may itself be extended to infer empirical yields for nucleosynthetic processes, although with the loss of neural net functionality and therefore speed.

Further improvements can be made by broadening the set of elements that is used, made possible by adding more nucleosynthetic channels, such as neutron-star mergers (Côté et al. 2017a) or sub-Chandrasekhar SNe Ia (Woosley & Kasen 2011; Shen et al. 2018). These will also give tighter constraints on the frequency of these additional channels. In observational contexts, we are limited to using only elements that do not undergo significant post-birth changes in abundance; inclusion of a model that maps the observed stellar elemental abundances to their birth abundances (e.g., Conroy et al. 2017) would allow a greater number of elements to be used. Furthermore, increasing nstars would allow us to add more free variables, for instance, SN Ia time-delay parameters, process-dependent outflow fractions, free solar abundances, and more complex (or hierarchical) star formation histories. The current precision of stellar age estimates does not seem to be a limiting factor for our method, especially because this is marginalized over, although more precise estimates would be expected to somewhat reduce the uncertainty on the SSP parameters.

When a set of stars is chosen for an analysis, it is important to consider the selection function (e.g., Haywood et al. 2016; Just & Rybizki 2016), and a study using only thin- or thick-disk stars may give us valuable insight into its effects. In our analysis of global SSP parameters, however, it appears to be sufficient to cover a large variety of the abundance space without the need for exhaustive knowledge of the selection function. This is in agreement with the work of Weinberg et al. (2019), who note that a given star’s abundances will carry the imprint of the global parameters and nucleosynthetic yields. Additional improvement may also be achieved by the use of Mg as the normalization element in the Chempy likelihood rather than Fe, as in Weinberg et al. (2019).

While this study has begun to explore the effects of modeling simplifications and incorrect yield tables, we caution that only a single set of analyses was run in each case, and the study is by no means intended as an exhaustive test to determine the applicability for the real MW. Other tailored tests will be necessary, for example, performing a detailed analysis of how chemical evolution modeling assumptions can bias the results (Côté et al. 2017b), or investigating the impacts of more complex subgrid physics in the hydrodynamical model, such as a metallicity-dependent IMF (Gutcke & Springel 2019).

7. Conclusions

In this paper, we have demonstrated a technique for inferring global galactic parameters controlling the SN Ia normalization, log_{10}(N_{\text{e}}), and the Chabrier (2003) IMF high-mass slope, $\alpha_{\text{IMF}}$, using only stellar chemical abundance and age data. This builds upon previous work by the extension to multiple stars, which requires a more sophisticated statistical model and sampling technique. The inference technique is both fast and flexible, allowing strong constraints to be placed on global parameters using a large number of stars in a few tens of CPU hours.

Our core model has been the flexible leaky-box GCE code Chempy (R17), which was used to predict elemental abundance ratios that are compared to observational data in a Bayesian framework. The Chempy model requires input parameters describing both global and local physics, with the latter being specific to a star’s formation environment. Forming a statistical model for multiple stars has thus required each star to carry its own set of ISM parameters, all of which must be marginalized over. The stellar birth time was treated as an extra free parameter, which was also marginalized over given some initial estimate. In addition, we included a model error parameter for each chemical element, which can account for inaccuracies in Chempy, for example, from incorrect chemical yield tables. This allowed the sampler to dynamically downweight elements that fit the data less well, reducing the bias in the global parameter estimates.

To allow for efficient sampling of the many-star posterior function, Chempy was replaced by a neural network, trained to reproduce output chemical abundances given some initial parameter set (see P18). This converts Chempy into a simple and differentiable analytic matrix function allowing us to use modern statistical methods to sample the high-dimensional posterior, in this case, Hamiltonian Monte Carlo methods (Neal 2012). The full analysis pipeline has been made publicly available with a comprehensive tutorial (Philcox & Rybizki 2019).21

Our analysis routine was tested using mock data; first with a data set computed by Chempy to test the neural network and sampling, augmented with broad observational errors of 5% (20%) in abundance (age). As the number of stellar data points, nstars increased, the estimated values of the SN Ia normalization and IMF slope were found to converge to the true values at high precision (≤1% for individual data sets with nstars ≥ 50). When few stars were used, we observed significant sample variance in the derived parameter estimates between data sets, indicating that caution must be used when interpreting inference results in single-star analyses such as R17.

To explore the bias created by assuming incorrect chemical yields, we similarly analyzed a data set created with a different set of yield tables, which was shown to give a bias of ~3% (~8%) in the posterior parameter estimates when model errors were (were not) included. This bias can be lowered by only using elements that are well predicted by our yield tables. Elements with larger model errors broadly corresponded to those with greater discrepancies between the yield tables, showing the utility of model errors as a diagnostic tool for determining how well model yields represent the chemistry of the Universe. In applications of this method to observational data, the analysis can be repeated with several different sets of yield tables to determine the bias empirically.

Using a mock data set drawn from a MW-like galaxy in the IllustrisTNG (Pillepich et al. 2018b) simulation (which has known values of the global parameters and yields), we were able to test the bias in the parameter estimates caused by the ISM physics simplifications in Chempy. These assumptions cause the outputs of Chempy to span only a limited subset of abundance space; a point outside the typical Chempy range may thus be expected to bias the inference results. In practice, this was found to be insignificant, with posterior parameter estimates consistent with the true values across the range of data-set sizes tested. For nstars = 100, we obtained constraints

---

21 github.com/oliverphilcox/ChempyMulti
of $\alpha_{\text{IMF}} = -2.283 \pm 0.010$ (statistical) $\pm 0.006$ (sample) and $\log_{10}(N_{\text{HI}}) = -2.889 \pm 0.011$ (statistical) $\pm 0.004$ (sample) compared to true values of $-2.3$ and $-2.9$, respectively. This is highly competitive when compared to canonical galactic parameter studies such as star counts in M31, which give $\alpha_{\text{IMF}} = -2.45^{+0.06}_{-0.03}$ (Weisz et al. 2015).

The model errors showed the metal abundance ratios to be highly consistent between IllustrisTNG and Chempy, but there were large discrepancies for [Fe/H] and [He/Fe], suggesting that Chempy is a relatively poor estimator of the overall metallicities (likely caused by our assumptions that AGB and SNe have the same feedback fraction to the local ISM and the feedback is accessible to new star formation immediately) though large model errors meant that these elements did not contribute significantly to the overall likelihood. We note that our inference was not able to place strong constraints on stellar ages; this can be improved by using a greater number of elements in the analysis.

The natural extension of this is the application to real data sets, for example, to red giant abundances from the APOGEE survey (Majewski et al. 2016), combined with stellar age priors (e.g., Ness et al. 2016). The statistical model remains the same in this context, but we are subject to a number of sources of uncertainty, which, while partially ameliorated by our model error parameters, can bias our inference. As shown above, the choice of chemical elements and yield tables is of paramount importance, and one may make guided choices from studies such as Weinberg et al. (2019) and P18, respectively. (Note also that we can obtain much stronger constraints on yield tables by using abundances from multiple stars, combining the techniques of P18 with this work.) Furthermore, because we can only observe current stellar abundances, there can be biases due to post-birth changes in chemical abundances (significantly affecting elements such as C and N). Additionally, although the physics simplifications made by Chempy were not found to have a large impact upon the TNG parameter constraints, this is not guaranteed for the real Universe. We are also sensitive to changes in the stellar lifetime functions and missing nucleosynthetic channels (e.g., neutron-star mergers).

These setbacks notwithstanding, it is clear that in tandem with additional constraints such as star counts (e.g., Weisz et al. 2015; Hosek et al. 2019), the methods in this paper could be used to obtain strong constraints on crucial galactic parameters such as the high-mass slope of the ISM and the number of SN Ia in the galaxy. Using approximate sampling methods such as ADVI, analysis with $n_{\text{stars}} \sim 1000$ will become possible, allowing us to rigorously exploit the huge volumes of chemical abundance data available. This will enable many probes of galactic physics, for example, testing the metallicity dependence of the IMF and attempting to infer the yield tables themselves.

We thank the following people for fruitful discussions: Robert Grand, Hans-Walter Rix, Rahul Dave, Chris Buswicka, and Henry Wang. We are grateful to the anonymous referee for a detailed report that helped improve the clarity and impact of this work. O.H.E.P. acknowledges funding from the Herchel-Smith foundation. J.R. acknowledges funding by the DLR (German space agency) via grant 50 QG 1403.

Software: ChempyMulti (Philcox & Rybizki 2019), ChempyScoring (Philcox & Rybizki 2018), Chempy (Rybizki et al. 2017b), scikit-learn (Pedregosa et al. 2011), astropy (Astropy Collaboration et al. 2013; Price-Whelan et al. 2018), PyMC3 (Salvatier et al. 2016), theano (Al-Rfou et al. 2016), corner (Foreman-Mackey 2016) & TikZ Bayesnet (github.com/jluttine/tikz-bayesnet).

Appendix A
Neural Network Implementation
We here discuss the specifics of the neural network used in this analysis, which was introduced in Section 3. The functional form is given by

$$h = W_0 \cdot x + b_0$$
$$y = W_1 \cdot f(h) + b_1$$

for input vector $x$ (dimension $n_{\text{in}}$), output vector $y$ (dimension $n_{\text{out}}$), and weights $\{W_i, b_i\}$, which are set via an optimizer during the network training. $h$ represents the hidden layer: a length $n_{\text{neuron}}$ vector that is transformed by some vector-valued activation function $f$ before the output is constructed, allowing the model to represent nonlinear functions. It is here chosen as a tanh function.

There are a total of six inputs to the Chempy function, from the global, local, and birth-time parameters, as stated in Table 1. To allow for more accurate network fitting, we augment the input parameter vector with the value of $T_7$ (giving $n_{\text{in}} = 7$), which is useful because Chempy has a most complex dependence on $T_7$. Instead of creating a single large network with $n_{\text{out}} = n_{\text{el}}$ outputs, we here construct $n_{\text{el}}$ individual networks with $n_{\text{out}} = 1$, allowing each element to be fit independently, giving greater network flexibility at smaller $n_{\text{neuron}}$. This requires little additional computation time because the networks can be trained in parallel, and initial testing showed $n_{\text{neuron}} = 40$ to give sufficient network accuracy without overfitting. For later efficiency, the $n_{\text{el}}$ fully connected networks are combined into a single sparsely connected network (with a total of $n_{\text{el}}n_{\text{neuron}}$ hidden-layer nodes), as illustrated in Figure 6.

To teach the network to emulate Chempy, we require a large volume of training data; sets of input parameter vectors and associated output Chempy abundances. Although a single run of Chempy at a given output time $T_7$ already computes elemental abundances at 28 equally spaced time steps, it is not pertinent to use these as 28 individual training points because the resolution is low for the first few time steps. Instead, we compute the model in full for each value of $T_7$ and take the final elemental abundances as training data, using a time step of $T_7/28$. The training data set is created from $1 \times 10^6$ random points in the six-dimensional parameter space (of $\Lambda, \Theta$, and $T_7$), with the SSP and ISM parameters being drawn from Gaussians (truncated for $\log_{10}(\text{SFR}_{\text{peak}})$ as in Section 2.2) centered at the prior-mean with $2\sigma_{\text{prior}}$ width (see Table 1). $T_7$ is drawn from a uniform distribution in $[1, 13.8]$ Gyr, ensuring good coverage over the relevant parameter space. This is the most computationally intense part of the analysis, with such a training set taking

23 In P18, we created training data via a regular grid in parameter space. The new approach was found to give a faster converging network and was thus adopted here.
Figure 6. Cartoon indicating the sparse neural network structure used in this analysis. We show a mock network with \( n_{\text{in}} = 2 \) input nodes \( \{x_i\} \) (representing Chempy parameters) and \( n_{\text{out}} = 2 \) output nodes \( \{y_j\} \) (representing element abundances). Although there appear to be six hidden-layer nodes (shown in gray), the \( j \)th output node is connected to only \( n_{\text{neuron}} = 3 \) hidden-layer nodes (labeled \( h_{1,1}, h_{1,2}, h_{1,3} \)), thus this structure is identical to a set of \( n_{\text{out}} \) fully connected networks with only a single output node and \( n_{\text{neuron}} = 3 \). In the full analysis, we use \( n_{\text{in}} = 7 \) (including a \( T^4 \) term), \( n_{\text{neuron}} = 40 \) and \( n_{\text{out}} = 8 \), embedded in a similarly sparse structure, which was found to give better accuracy than a single fully connected network. Cartoon created using TikZ Bayesnet.

\(~200\) CPU hours to compute on a modern desktop machine, but can be trivially parallelized. For improved fitting, all network inputs and outputs are standardized, with the new values \( \hat{p}_i \) being derived from their unstandardized forms \( p_i \) via

\[
\hat{p}_i = \frac{p_i - \mu_i}{\sigma_i},
\]  

where \( \mu_i \) and \( \sigma_i \) are the mean and standard deviation of \( p_i \). The uniformly distributed \( T_i \) is instead linearly mapped to the interval \([0, 1]\). This gives a total of \( n_{\text{neuron}}(n_{\text{in}} + n_{\text{out}} + 1) + n_{\text{out}} = 361 \) free weight parameters for each of the \( n_{\text{el}} \) networks, which are found by training with an Adam optimizer (Kingma & Ba 2014), using a mean-square-error (L2) loss function and an adaptive learning rate, reducing as the training loss plateaus. This was implemented using the scikit-learn package (Pedregosa et al. 2011) in Python, with training taking \(~1\) CPU hours (but may be parallelized \( n_{\text{el}} \)-fold).

Figure 7. Absolute deviation between neural network predictions and true Chempy abundances for eight elements, computing distances from \( 5 \times 10^4 \) parameter space samples, with inputs drawn from Gaussians centered at the Chempy priors (Table 1) with widths of \( 2\sigma_{\text{prior}} \). We show the median and 16th/84th percentile deviations for two network configurations, using a single network for each element (blue) and using a joint network for all elements (red). Both instances are trained with \( 10^6 \) data points using \( n_{\text{neuron}} = 40 \), and the former gives superior results.

Testing is performed by comparing the true abundances to the neural network predictions across an independent test set of \( 5 \times 10^4 \) points (each consisting of an input parameter vector and a set of output abundances), computed as for the training data. Using the L1 distance metric (the absolute deviation between two values), we obtain a median error of \( 0.005^{+0.008}_{-0.004} \) dex across the entire testing parameter space and \( n_{\text{el}} = 8 \) elements, well below typical observational errors of 0.05 dex, thus we take the network to be a good approximator of the Chempy function. Figures 7 and 8 show the error as a function of the element and position in parameter space, respectively, with the former also demonstrating the benefits from using individual networks for each element rather than a single fully connected network. As expected, the network errors are small in the center of the distribution, but grow toward the edges of parameter space, where the function is sampled less finely. In particular, errors are greatest at the extremes of \( T_{\text{star}} \); for this reason, we exclude stars with \( T_{\text{star}} \notin [1, 13.8] \) Gyr from the analysis, avoiding the need for a greater volume of training data. If we required a more accurate network, this could be obtained using a large training data set (possibly encompassing a greater prior width) or more neurons.
Appendix B
Introduction to Hamiltonian Monte Carlo

We here present a broad overview of the HMC algorithm, which allows us to sample relatively high-dimensional posteriors with much greater efficiency than standard MCMC methods. In this paper, HMC is implemented via the PyMC3 package (Salvatier et al. 2016).

Following the notation of Betancourt & Girolami (2013), consider a posterior distribution $\pi(q)$ with parameter $q$, from which require samples. Instead of sampling $\pi(q)$ directly, we here introduce a momentum parameter $p$ and sample the joint density $p(p,q)\pi(q)$, for user-defined conditional distribution $p(p|q)$ (often chosen as a Gaussian). In line with classical mechanics, we introduce a Hamiltonian density

$$H(p, q) = -\log \pi(p, q) = T(p|q) + V(q),$$

identifying the kinetic and potential energies $T(p|q) = -\log \pi(p|q)$ and $V(q) = -\log \pi(q)$, respectively. (The kinetic energy becomes a simple quadratic in $p$ if we choose a Gaussian for $\pi(p|q)$.)

Given this identification, we sample a value of the momentum $p$ from the conditional distribution $p(p|q)$, then evolve the variables $p$ and $q$ for some period of time according to Hamilton’s equations for $H(p, q)$,

$$\frac{dq}{dt} = \frac{\partial H}{\partial p}, \quad \frac{dp}{dt} = -\frac{\partial H}{\partial q},$$

requiring solution of a first-order differential equation (usually via leapfrog methods). After some number of time steps, a new value of $p$ is drawn and the process repeated, with the individual samples of $q$ at each time step forming the posterior chain. This results in a much more efficient sampling of the parameter space than just making random jumps in $q$ (as in conventional MCMC algorithms) because we additionally use the gradients of $H$ with respect to $p$ and $q$. Notably, this requires differentiability of the posterior $\pi(q)$, which limits the utility of HMC in many astrophysical contexts.

One pitfall of HMC is the addition of multiple free parameters that control the number and size of integration steps that should be taken from a given starting $(p, q)$ before a new momentum $p$ is drawn, which could require difficult tuning. This is solved with the no U-turn sampler (NUTS; Hoffman & Gelman 2011), which (a) provides a physically motivated way in which to compute the step size and (b) finds the optimal number of integration steps by integrating Hamilton’s equations both forward and backward in time until the path in phase-space doubles back on itself (and hence stops

Figure 8. Mean neural network error across all elements as a function of position in the six-dimensional Chempy parameter space. The histograms on the diagonal show the distribution of test data points, with their colors indicating the mean error in each bin. Full (dashed) lines indicate the median ($\sigma_{\text{train}} = 2\sigma_{\text{prior}}$) training values used in this sample. Off-diagonal plots show the marginal distribution of the error with respect to pairs of parameters. (Note that the $x_{\text{out}}$ parameter is restricted to [0, 1], as in the full analysis, because values outside this region are unphysical.) The network errors are small in the center of the parameter space (where the priors are concentrated), giving minimal bias to the inference.
Although HMC provides a large reduction in computation time compared with standard MCMC approaches, we can still encounter difficulties for very complex or high-dimensional posteriors, with the sampler taking too long to converge. For the analysis presented above, when we restricted ourselves to sampling times shorter than a few hours, we were limited to \( n_{\text{stars}} \lesssim 200 \), although we were still able to produce high-precision parameter estimates with this size of data set.

For more efficient sampling with large \( n_{\text{stars}} \), it may be more appropriate to use an HMC-within-Gibbs sampling approach, with HMC used to perform the parameter updates for \( \Lambda, \Sigma \), and \( \{\Theta, T_i\} \) separately, (as suggested in Neal 2012), although this has not been implemented here. As mentioned above, an additional possibility is to use approximate sampling methods such as automatic differentiation variational inference (ADVI; Kingma & Welling 2013; Kucukelbir et al. 2016; Roeder et al. 2017), which approximates the (possibly transformed) posterior function as a product of univariate Gaussians that can be trivially sampled from. This approximation depends on a number of latent parameters (describing the shape and location of each Gaussian), which are optimized via gradient descent, again requiring differentiability. While the assumption of Gaussianity may seem to be highly restrictive, it is often found to work well in practice, especially when we additionally allow for correlations between some or all parameters (in “Full Rank” ADVI, in contrast to the standard “Mean Field” ADVI).

While not considered in this paper, this may be useful for analyses containing a greater number of model parameters, for instance, if the chemical yields are also left free.

### Appendix C

**Full Global Parameter Corner Plot**

Figure 9 shows the corner plot of the *Chempy* posterior for HMC sampling of the TNG data set using \( n_{\text{stars}} = 200 \), as discussed in Section 6.3. Because the full posterior exists in a 810-dimensional space, we show only the portions corresponding to the SSP parameters, \( \Lambda \), and model errors, \( \Sigma = \{\sigma_i^{\text{model}}\} \). While the \( \log_{10}(N_{\text{Ia}}) \) parameter is highly consistent with the true value, there is a slight tension in the \( \alpha_{\text{IMF}} \) parameter, although this may be caused by sample bias. The large nonzero errors of \([\text{Fe}/\text{H}] \) and \([\text{He}/\text{Fe}] \) (here denoted by \( \sigma_{\text{Fe}} \) and \( \sigma_{\text{He}} \)) are clearly apparent, with the model error histograms matching those of Figure 3 and often close to the prior half-Cauchy distributions. Furthermore, we note strong correlations between \( \alpha_{\text{IMF}} \) and \( \log_{10}(N_{\text{Ia}}) \) (matching that found in R17), with a larger \( \alpha_{\text{IMF}} \) leading to more SN II, which require more SN Ia to obtain the correct abundance ratios of \( \alpha \) and iron-peak elements. The model errors appear to be largely uncorrelated both with each other and with the SSP parameters, although there is weak correlation between \( \sigma_{\text{Fe}} \) and \( \sigma_{\text{He}} \) because both trace the overall metallicity of the simulation.
ORCID iDs
Oliver H. E. Philcox @ https://orcid.org/0000-0002-3033-9932
Jan Rybizki @ https://orcid.org/0000-0002-0993-6089

References
Abate, C., Pols, O. R., Izzard, R. G., & Karakas, A. I. 2015, A&A, 581, A22
Al-Rfou, R., Alain, G., Almabairi, A., et al. 2016, arXiv:1605.02688
Andrews, B. H., Weinberg, D. H., Johnson, J. A., Bensby, T., & Feltzing, S. 2012, AcA, 62, 269

Figure 9. Corner plot illustrating part of the sampled posterior function using 10^4 posterior samples obtained using HMC methods. We display only portions corresponding to the global SSP parameters, \(A = (\alpha_{\text{IMF}}, \log(N_d))\), and model errors for each element \(S = \{s_{\text{model}}\}\). The true values of \(A\) are marked in blue and are highly consistent with the SN Ia parameter, with a slight offset observed for \(\alpha_{\text{IMF}}\). Dashed lines in the one-dimensional histograms indicate the 16th, 50th, and 84th percentiles, and smoothed contours (at 1–4σ levels) are shown in the two-dimensional histograms. The prior distributions are indicated by red curves in the histograms. Plot created using corner (Foreman-Mackey 2016).
