



Disentangled Representation Learning for Astronomical Chemical Tagging

Damien de Mijolla¹ , Melissa Kay Ness^{2,3} , Serena Viti^{4,1} , and Adam Joseph Wheeler² 

¹Department of Physics and Astronomy, University College London, Gower Street, WC1E 6BT, UK; ucapdde@ucl.ac.uk

²Department of Astronomy, Columbia University, Pupin Physics Laboratories, New York, NY 10027, USA

³Center for Computational Astrophysics, Flatiron Institute, 162 Fifth Avenue, New York, NY 10010, USA

⁴Leiden Observatory, Leiden University, P.O. Box 9513, 2300 RA Leiden, Netherlands

Received 2020 November 18; revised 2021 February 28; accepted 2021 March 3; published 2021 May 18

Abstract

Modern astronomical surveys are observing spectral data for millions of stars. These spectra contain chemical information that can be used to trace the Galaxy’s formation and chemical enrichment history. However, extracting the information from spectra and making precise and accurate chemical abundance measurements is challenging. Here we present a data-driven method for isolating the chemical factors of variation in stellar spectra from those of other parameters (i.e., T_{eff} , $\log g$, $[\text{Fe}/\text{H}]$). This enables us to build a spectral projection for each star with these parameters removed. We do this with no ab initio knowledge of elemental abundances themselves and hence bypass the uncertainties and systematics associated with modeling that rely on synthetic stellar spectra. To remove known nonchemical factors of variation, we develop and implement a neural network architecture that learns a disentangled spectral representation. We simulate our recovery of chemically identical stars using the disentangled spectra in a synthetic APOGEE-like data set. We show that this recovery declines as a function of the signal-to-noise ratio but that our neural network architecture outperforms simpler modeling choices. Our work demonstrates the feasibility of data-driven abundance-free chemical tagging.

Unified Astronomy Thesaurus concepts: [Stellar astronomy \(1583\)](#); [Neural networks \(1933\)](#); [Stellar spectral lines \(1630\)](#)

1. Introduction

Galactic archeology, the subfield of astronomy interested in reconstructing the Galaxy’s history, has recently experienced substantial growth. This has been spurred by stellar surveys such as RAVE, APOGEE, GALAH, LAMOST, Gaia, and Gaia-ESO (Steinmetz et al. 2006; Cui et al. 2012; Gilmore et al. 2012; Randich et al. 2013; De Silva et al. 2015; Majewski et al. 2017; Gaia Collaboration et al. 2018b). These surveys have obtained spectra and, in the case of Gaia, astrometry and photometry for hundreds of thousands to millions of stars across the Galaxy. These data have enabled measurement of stellar abundances, distances, and ages across the Galaxy. Future missions are also on the horizon (Bonifacio et al. 2016; de Jong et al. 2016; Tamura et al. 2016; Kollmeier et al. 2017).

Chemical element abundances derived from stellar spectra are core to archeological pursuits. While there are evolutionary and environmental factors that can impact the surface abundance of a star, the stellar siblings originating from molecular clouds share similar chemical fingerprints, and abundances can be used to link stars to individual molecular clouds (Feng & Krumholz 2014; Bovy 2016; Ness et al. 2018; Krumholz et al. 2019; Liu et al. 2019). The chemical space of stars in the Milky Way’s disk seems fairly low-dimensional, with stars born at the same radius and time being chemically similar or even identical within measurement precision (Ting et al. 2012; Ness et al. 2019; Price-Jones & Bovy 2019; Weinberg et al. 2019). Indeed, at solar metallicity, the APOGEE survey shows that 1% of field stars are as chemically similar as stars that are known to be from the same individual birth cluster (Ness et al. 2018). This doppelganger rate alone renders chemical tagging of stars to their individual birth sites using ≈ 20 abundances alone rather difficult. Nevertheless, identifying chemically identical or near-identical stars has high utility in reconstructing the galaxy’s formation, for example, in

estimating the number of star-forming clusters in the galactic disk (e.g., Ting et al. 2016; Kamdar et al. 2019) or for understanding how stars have moved over time (e.g., Beane et al. 2018; Coronado et al. 2020; Price-Jones et al. 2020; Frankel et al. 2018). Furthermore, detailed abundances allow for connecting stars to their birth radii, as well as their time of formation (Bedell et al. 2018; Feuillet et al. 2019; Ness et al. 2019; Casali et al. 2020). Typically, efforts to identify chemically identical stars have involved estimating surface abundances by comparing observations to synthetic spectra and then running a clustering algorithm (Hogg et al. 2016; Price-Jones & Bovy 2019). This procedure is hampered by its reliance on imperfect stellar models to obtain the abundance labels that describe the spectra. Typically employed 1D non-LTE stellar simulations do not fully capture the complexity of stellar photospheres. Often, only a fraction of the spectrum (the locations of a subset of cleanly identified features) is utilized. There may also be systematic abundance offsets in the derived abundance labels due to signal-to-noise ratio (S/N) dependencies of their derivation or unmodeled instrumental imprints on the spectra, meaning that abundance estimates are subject to artifacts (e.g., Holtzman et al. 2015). Data-driven approaches have provided higher-precision abundances for stars across surveys (e.g., Ness et al. 2015; Casey et al. 2017; Ho et al. 2017; Wheeler et al. 2020). However, at their core, these approaches still rely on stellar models to provide stellar parameter and abundance labels for the training data.

In this paper, we demonstrate the feasibility of identifying chemically identical stars without explicit use of measured abundances. We apply a neural network with a supervised disentanglement loss term to a synthetic APOGEE-like data set of spectra. The model learns a representation of spectra that traces abundances independently from the nonchemical factors of variation. That is, it controls for changes in the spectra

caused by, for example, effective temperature, T_{eff} , and surface gravity, $\log g$. This isolates the chemical variation expressed in the spectra. Stars with identical chemical compositions but differing T_{eff} and $\log g$ are mapped to nearly identical representations.

Unlike approaches based on explicit abundance estimates, this model naturally exploits the full available wavelength range, including blended lines, to effectively estimate the chemical composition. Additionally, it does not depend on stellar models and so does not suffer from associated systematics. We find that the learned low-dimensional representation of synthetic spectra can be transformed linearly into abundances with high precision.

Our method relies on the assumption that there is no correlation or statistical dependency between the physical and chemical factors of variation. Although such an assumption has been used (Valenti & Fischer 2005; Jofré et al. 2019), stellar processes, such as atomic diffusion and dredge-up, contribute to modifying surface abundances away from their birth values (Dotter et al. 2017). Because our model learns a representation of stellar spectra in which all variation dependent on nonchemical parameters is removed, assuming evolutionary changes in abundances correlate with the nonchemical factors we parameterize, the effect of these processes should also be removed from the representation, meaning that it will reflect birth, rather than present-day, abundances.

Studies of open cluster populations have demonstrated that stars can change in their element abundances by 0.1–0.3 dex across the main sequence to the giant branch (Bertelli Motta et al. 2018; Souto et al. 2019). This is also in line with theoretical expectations and a consequence of physical processes like atomic diffusion (Dotter et al. 2017). It is therefore a relevant and important distinction that we interrogate the spectra of a star for its birth abundance composition, as opposed to its present-day composition.

We have structured our paper such that our technical work on supervised disentanglement, of potential interest outside the astronomy community, is presented separately from our astrophysical application. After introducing the associated literature in Section 2, in Section 3, we discuss disentangled representation learning. In Section 4, we adapt this method for chemical tagging and show our experimental results on an APOGEE-like data set, demonstrating the recovery of chemically identical stars in the presence of noise. We also compare our approach to a baseline method (Price-Jones & Bovy 2019). We finish by discussing in Section 6 some important aspects of our method that are not explored using synthetic data that comprise the next steps, as well as our method’s benefits.

2. Related Work

2.1. Disentangled Representation Learning

There is a growing body of literature on using neural networks for learning to encode data into interpretable representations. Unsupervised disentanglement methods, such as beta-VAE (Higgins et al. 2017) and infogan (Chen et al. 2016), attempt to find representations in which distinct informative factors of variation (such as lighting conditions and object orientation in the context of images) are encoded in separate dimensions (Bengio et al. 2013). However, recent results suggest that finding such disentangled representations in a fully unsupervised setting is fundamentally ill posed without

additional assumptions or priors being set (Locatello et al. 2019).

Supervised disentanglement methods (Schmidhuber 1991; Ganin et al. 2016; Lample et al. 2017) specify labels for factors of variations that should be excluded from the learned representation. They aim to find a representation of inputs in which the specified factors of variation are removed from the representation but for which all other factors of variation are still present. A perfectly disentangled representation is statistically independent from the specified factors of variation. However, there will often be a trade-off between disentanglement and reconstruction (Lezama 2019).

Supervised disentangled learning has primarily been implemented through an adversarial training scheme, in which an autoencoder—a neural network with a lower-dimensional bottleneck that is trained at reconstructing inputs—learns to encode its input in such a way that a second network is unable to predict the to-be-disentangled labels from the encoded representation (Edwards & Storkey 2016; Lample et al. 2017; Hadad et al. 2018). It has also been proposed to obtain a disentangled representation by enforcing that an autoencoder learn a representation in which latent and labels are factorized. This has been done within the variational autoencoder framework in Louizos et al. (2016) but also with adversarial autoencoders in Polykovskiy et al. (2018). Another existing avenue for obtaining supervised disentanglement can be found through a cyclic training scheme that encourages the latent to remain unchanged after reencoding outputs obtained after modifying the factors of variation. This approach has been demonstrated in the context of variational autoencoders in Chen et al. (2019) and Jha et al. (2018).

Supervised disentanglement could be a very useful technique in the field of astronomy, and we hope that this paper will be beneficial for showcasing its potential. For example, supervised disentanglement could be used in astronomical calibration to remove the effects of individual fibers or weather conditions on spectra. This could be done by learning a representation that is, for example, statistically independent from the fiber number for a multi-object spectrograph. Such an approach would be complimentary to our paper, as our proposed method requires precisely calibrated spectra and additional augmentation to handle systematic artifacts.

2.2. Data-driven Chemical Tagging

Chemical tagging describes the reconstruction of individual cluster groups via abundance information (Freeman & Bland-Hawthorn 2002; Ting et al. 2016; Casey et al. 2019). The concept has extended to the identification of chemically anomalous stars of particular formation origins (Hogg et al. 2016; Schiavon et al. 2017), the association of and differentiation between stellar groups and populations using abundances (Martell et al. 2016; Hawkins & Wyse 2018; Simpson et al. 2019), and grouping stars by chemical similarity (e.g., Price-Jones & Bovy 2019). Recent work indicates that there is limited feasibility of chemically tagging stars back to their individual cluster origins using the ≈ 20 individual abundance measurements alone from resolution $R = 22,500$ spectra (e.g., Ness et al. 2018). Most approaches use the labels that describe the spectra, and new approaches have improved the precision of these labels (Ness et al. 2015; Leung & Bovy 2018; Ting et al. 2019). Novel approaches to chemical tagging include those presented in Blanco-Cuaresma & Fraix-Burnet (2018)

and Jofré et al. (2017), which used techniques from the field of phylogeny, and Price-Jones & Bovy (2019), who identified chemically identical stars without explicit use of abundances. The concurrent work presented in O’Brian et al. (2021) uses a machine-learning algorithm loosely similar to ours for improving stellar abundance estimation.

The method proposed in Price-Jones & Bovy (2019), which itself expands upon earlier work presented in Price-Jones & Bovy (2017), bears some clear similarity to our work in that it uses a data-driven model applied directly to spectra to learn a representation in which undesirable parameters are removed. They fit a polynomial model of the nonchemical parameters to every single wavelength bin. The residuals of this fit were then considered to only contain chemical information. They then ran a clustering algorithm on a compressed representation of the residuals obtained after principal component analysis to identify chemically similar groups. However, as discussed in their paper, this method comes with some limitations. A polynomial fit may not be an optimally flexible functional form, particularly across a breadth of stellar evolutionary states (see, for example, Ting et al. 2019). As such, it is unlikely to perfectly remove the physical parameters of variation from the residuals. Furthermore, by fitting nonchemical parameters in isolation, any joint dependencies between chemical and nonchemical factors of variation on spectral line strengths are ignored.

3. Methods

We present here an overview of our method. Section 3.1 introduces our underlying assumptions on the data-generating process, the problem we are trying to answer, and the broad strokes of our method. In Section 3.2, we dive deeper and present a neural network architecture for solving our introduced problem. In Section 3.3, we present two different methods of enforcing a disentangled representation, a key component in our method.

3.1. Problem Statement

We consider a setup in which a data set $X = \{x_1, \dots, x_n\}$ is observed. We assume the data set to be generated deterministically from latent variables through a mapping unknown to us. Despite not knowing this mapping, we assume that a subset of the latent variables can be accurately estimated. As such, we can subdivide latent variables into a vector of known variables u and a vector of unknown variables v . For our method to work, we further assume that u and v are (marginally) statistically independent (i.e., $p(v|u) = p(v)$). This corresponds to the notion that u and v cannot be used to predict each other.

In this paper, we present a general method for quantifying the similarity of observations x as measured in terms of unknown variables v . In particular, our method provides a means for identifying observations x sharing an identical or near-identical vector v without knowledge of the mapping from latent to observed variables.

Our method learns a mapping, parameterized by a neural network, from observations x to a vector z acting as a proxy for unknown variables v . More precisely, we learn a mapping such that observations sharing a common parameterization for v , in turn, share a near-identical representation for z .

We achieve this through finding a representation z that is statistically independent from the known and provided

parameters, u , but when combined with these known parameters, it is capable of perfectly reconstructing the observations x . This ensures that our latent variables contain all of the information contained within the unknown variables v but no additional superfluous information.

How does this assumed setup relate back to astronomical chemical tagging? For chemical tagging, we have access to the stellar spectra of stars, x , from which we seek to identify stars sharing an identical chemical composition, v . Although we are capable of estimating physical parameters, u , fairly accurately, shortcomings in spectral synthesis make it difficult to relate spectra back to their chemical composition.

3.2. Approach

We rely on a conditional autoencoder, a type of neural network, to learn the mapping to the lower-dimensional representation z . Our autoencoder (represented in Figure 1) is composed of two separate neural networks. A conditional encoder takes as inputs the observations, x , concatenated with known parameters, u , and returning a latent representation, z (for the remainder of the paper, we adopt machine-learning terminology and refer to z as latents), and a conditional decoder takes z and u as inputs and is trained to output reconstructed observations x .

This autoencoder is trained to minimize the following loss function:

$$L_{\text{AE}} = L_{\text{rec}} + \lambda L_{\text{dis}}, \quad (1)$$

where L_{rec} is a reconstruction loss. In our experiments, we used the mean squared loss,⁵

$$L_{\text{rec}} = E_{(x,u) \sim p(x,u)} [\|D(E(x, u), u) - x\|_2^2], \quad (2)$$

The L_{dis} is a disentanglement loss, acting to ensure that the latent, z , is maximally disentangled from the known and provided parameters, u ; λ is a term controlling the trade-off between reconstruction and disentanglement. The disentanglement loss is there to push the network toward learning a latent representation that is statistically independent from the observed parameters, u , and should be minimal when z is independent from u . We present two formulations of L_{dis} in Section 3.3.

During training, the autoencoder is iteratively shown data points grouped into batches—subsets of the data set. The autoencoder’s loss, as described above, is evaluated on each batch, and the derivatives of this loss with respect to the neural network parameters are used to update the parameters in the direction minimizing the loss function. After training, the neural network will have converged to parameterizing a mapping that (locally) minimizes the loss function. Although not a global minimum, the learned mapping, in part because of the stochastic nature of the training process, will typically be a good minimizer of the loss function.

Our neural network, in minimizing the loss function described by Equation (1), simultaneously minimizes the reconstruction and disentanglement terms with a trade-off controlled by λ . Minimizing the disentanglement loss term corresponds to learning a latent representation statistically independent from factors of variation parameterized by u . This is achieved by removing all related information from the latent. The reconstruction term will be minimized when z and u are

⁵ $\|x\|_2 := \sqrt{x_1^2 + \dots + x_n^2}$.

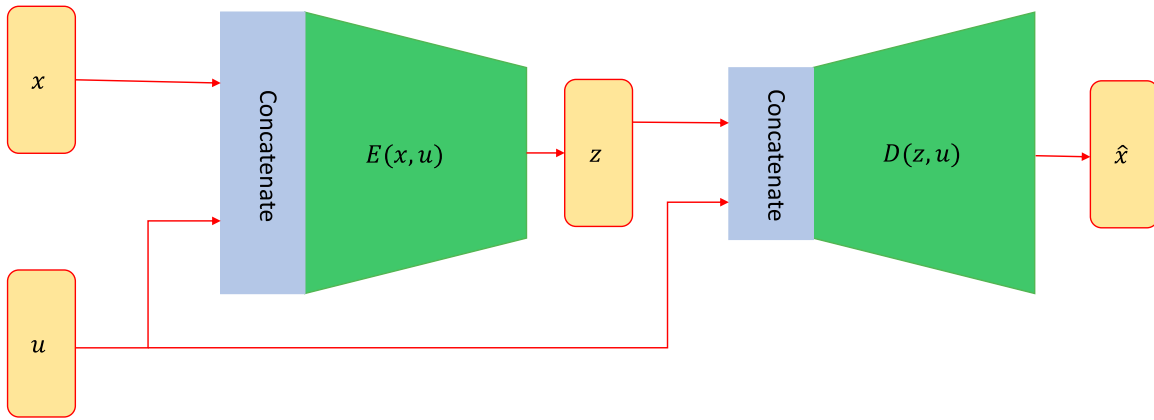


Figure 1. Diagram of the conditional autoencoder architecture. We denote the reconstructed observation as \hat{x} . For chemical tagging, x corresponds to stellar spectra and u to the physical factors of variation.

sufficient for reconstructing observations x . Combined, these two loss terms will be minimized when all of the information required for modeling the observations x not included within u is contained within the latent z . While it may not always be possible to minimize both loss terms together, we know that it is possible to do so for data generated as described in Section 3.1. Indeed, a global minimum of the loss function would be reached for a neural network that encoded the observations x into v and decoded back to x .

In addition to isolating unknown factors of variation, v , we have found that, at least for the problems we have considered, supervised disentanglement maps observations with shared parameter values, v , to nearly identical latents, z . We attribute this to our set of assumptions (see Section 3.1). This property makes some intuitive sense when we take a moment to consider how our autoencoder might map the observations, x , generated from a common shared vector of unknown parameter values, v , but each with different values of the observed parameters, u . If the mapping does not project all of these observations to a common latent value, then the latent value, z , will be informative about the parameter value u (as some u are then more or less likely based on the observed z). Therefore, z and u will no longer be statistically independent.

In practice, our neural network will only approximately minimize our loss function and so will not perfectly map observations sharing common parameter values, v , to the same latent z . Observations sharing common parameter values will thus appear as overdensities in the latent space. These overdensities can then be identified, for example, by running a clustering algorithm such as K-means (Lloyd 1982) or finding those observations that are particularly close according to some distance metric. Alternatively, we can instead identify such observations in the data space if we use the decoder to convert all latents with a common set of parameters, u_i .

3.3. Implementation of Supervised Disentanglement

We present two alternative methods, FaderDis and FactorDis, for learning a disentanglement loss L_{dis} encouraging statistical independence. FaderDis is an adaptation of the Fader disentanglement architecture presented in Lample et al. (2017) modified for our purposes. FactorDis is, to our knowledge, a novel architecture for supervised disentanglement. We present here the architectures investigated.

3.3.1. Factor Disentanglement (FactorDis)

The FactorDis method enforces independence by training a critic network to differentiate between samples from the joint distribution $p(x, u, z)$ and samples in which the statistical dependency between z and u has been forcibly removed. Analogously to generative adversarial networks (Goodfellow et al. 2014), the conditional autoencoder is adversarially trained to generate samples that hinder the critic network’s ability to do its job.

The joint distribution $p(x, u, z)$ can be expressed using Bayes’ rule as

$$p(x, u, z) = p(x|u, z)p(u, z). \quad (3)$$

This can be rewritten as

$$q(x, u, z) = p(x|u, z)p(u)p(z) \quad (4)$$

if and only if u is statistically independent from z . If the joint distribution $p(u, z)$ is not factorizable, the distributions $q(x, u, z)$ and $p(x, u, z)$ will be different. It follows from this that u and z are statistically independent if and only if the samples (x, u, z) drawn according to $p(x, u, z)$ are indistinguishable from those sampled from $q(x, u, z)$.

How can we generate samples from these two distributions? If we consider our autoencoder to be an idealistic autoencoder capable of perfectly reconstructing its inputs, then the encoder and decoder can be viewed as respectively approximately parameterizing $p(z|u, x)$ and $p(x|u, z)$, which are both deterministic functions. We can thus draw samples from $p(x, u, z) = p(z|u, x)p(u, x)$ by first randomly sampling from the data set to obtain (u, x) and then using the encoder to obtain the associated z .

We can similarly draw samples from $q(x, u, z) = p(x|u, z)p(u)p(z)$ through reusing our samples drawn from $p(x, u, z)$. By scrambling the (u, z) pairs within a batch, we can effectively remove any joint information between u and z (Belghazi et al. 2018), which results in samples drawn from the marginal distribution $(u, z) \sim p(u)p(z)$. We can then use the decoder that approximates $p(x|u, z)$ to obtain approximate samples $q(x, u, z)$ drawn from $p(x|u, z)p(u)p(z)$.

As stated above, enforcing statistical independence is the same as finding a latent representation, z , such that samples drawn according to these two procedures are indistinguishable. This bears a strong similarity to the training objective of generative adversarial networks that attempt to train a generator

such that generated samples are indistinguishable from samples drawn from a data set. As such, we can take inspiration from existing generative adversarial network architectures to solve our disentanglement objective.

In generative adversarial networks (Goodfellow et al. 2014), a critic network is trained to distinguish between samples drawn from a data set and samples created by a generator network fed samples from a well-understood probability distribution. The generator and critic network are jointly optimized in a minimax game. That is, the critic attempts to maximally distinguish between the two data streams, and the generator attempts to minimize the critic network’s ability at doing so. The global optimum of this two-player game occurs when both the generator and critic network can no longer improve—when the two data streams are identical.

For our disentanglement neural network architecture, we take heavy inspiration from the Wasserstein generative adversarial network (Arjovsky et al. 2017). We use an architecture parallel to that of generative adversarial networks. However, instead of differentiating between real and fake samples, we differentiate between samples from $p(x, u, z)$ and $q(x, u, z)$ generated using the autoencoder. This leads to optimizing the following minimax objective:

$$\min_{\text{AE}} \max_{C \in \mathbb{D}} \mathbb{E}_{(x,u,z) \sim p(x,u,z)} [C(x, u, z)] - \mathbb{E}_{(x,u,z) \sim q(x,u,z)} [C(x, u, z)], \quad (5)$$

where \mathbb{D} is the space of 1-Lipschitz continuous function and $C(x, u, z)$ refers to a critic network that takes as input a vector in which observations x , latents z , and parameters u are concatenated and attempts to differentiate between the different type of samples generated by our autoencoder.

The critic network attempts to maximize Equation (5). In order to constrain the critic network to learning a Lipschitz continuous function, we add a gradient penalty term, weighted by a constant λ , to the loss, as was introduced in Gulrajani et al. (2017). This leads to a critic loss function

$$L_{\text{critic}} = \mathbb{E}_{(x,u,z) \sim q(x,u,z)} [C(x, u, z)] - \mathbb{E}_{(x,u,z) \sim p(x,u,z)} [C(x, u, z)] + \lambda \mathbb{E}_{(x,u,z) \sim r(x,u,z)} [(\|\nabla_{x,u,z} C(x, u, z)\|_2 - 1)^2], \quad (6)$$

where $r(x, u, z)$ is implicitly defined as sampling uniformly along straight lines between pairs of points sampled from the distributions $p(x, u, z)$ and $q(x, u, z)$. Further information about this sampling procedure can be found in Gulrajani et al. (2017).

Our autoencoder, which plays the role of a generator network, is trained to minimize Equation (5) while simultaneously minimizing the reconstruction loss function:

$$L_{\text{AE}} = L_{\text{rec}} + \lambda_2 \mathbb{E}_{(x,u,z) \sim p(x,u,z)} [C(x, u, z)] - \lambda_2 \mathbb{E}_{(x,u,z) \sim q(x,u,z)} [C(x, u, z)]. \quad (7)$$

This loss function combines the reconstruction loss that is traditionally used for optimizing autoencoders with a Wasserstein loss. In addition, unlike for generative adversarial networks, as both data streams are passed through the generator, they are both used for optimizing the generator. Training involves jointly minimizing the critic and autoencoder losses. The two different types of losses are weighted by a factor λ_2 . Experimentally, we found that it was crucial to correctly set the factor λ_2 , such that neither the reconstruction

term nor the disentanglement term in the loss dominated over the other.

3.3.2. Fader Disentanglement (FaderDis)

The FaderDis method of disentanglement follows the setup presented in Lample et al. (2017), in which an autoencoder is adversarially trained to learn a latent representation from which an auxiliary network is incapable of predicting u . Since the method is designed to operate on discrete variables, we discretize out the parameter space u into n equal-sized bins.

In this method, an auxiliary network, A , accepts latents, z , as inputs and returns as outputs a vector of size equal to the number of discretized bins. It is trained using a cross-entropy loss to predict the probability of the corresponding u vector falling in each of the n bins. The autoencoder is then trained alongside this auxiliary network. The autoencoder attempts to minimize the auxiliary network loss weighted by a factor λ_1 while also maximizing its own reconstruction loss. The autoencoder loss takes the form

$$L_{\text{AE}} = L_{\text{rec}} - \lambda_1 (E_{(x,u) \sim p(x,u)} [-u_n \log(A(E(x, u))))], \quad (8)$$

where u_n denotes the one hot-encoding vector after the discretization procedure, with the subscript n referring to the bin in which the parameters u fall.

The global optimum of this two-player minimax game will occur when the autoencoder learns to reconstruct observations using a latent z that does not contain any helpful information for the auxiliary network. Since the auxiliary network attempts to learn $p(u|z)$, this will occur when $p(u|z) = p(u)$ or, equivalently, u and z are statistically independent.

4. Application to Stellar Spectra

We wish to learn a representation of stellar spectra that disentangles factors of variation of interest (chemical abundances) from the observed parameters, u . We tested both methods with metallicity, $[\text{Fe}/\text{H}]$, as a known and unknown parameter ($u = [T_{\text{eff}}, \log g, [\text{Fe}/\text{H}]]$ and $u = [T_{\text{eff}}, \log g]$). After training, without any explicit knowledge of abundance labels, v , our neural network will find a mapping from observations, x , and parameters, u , to latents, z , such that stars sharing a common abundance are mapped to nearly identical latents.

We demonstrate our method using a synthetic data set described in Section 4.1. The data set is designed to mimic the spectral variability found within the APOGEE red giant sample. This allows us to carry out a proof of concept for our method in an ideal and controlled environment, in which independence between chemical and physical parameters is guaranteed and for which we were certain to have accounted for all factors of variation. This is an important first step in demonstrating the viability and performance of our method.

We quantify the performance of our generative model with a chemical abundance twin recovery test, comparing to simpler models, that also removes factors of variation T_{eff} , $\log g$, and $[\text{Fe}/\text{H}]$. We do this for a number of S/N qualities. We note that the performance of our method, in practice, will be sensitive to any calibrations or instrumental artifacts that are poorly modeled or not included as observed parameters. We also expect that the dimensionality of real data may be far lower than that of our synthetic library. This is because we do not restrict our realized abundances to the correlations observed in

Table 1
Ranges Used for Uniformly Sampling the Nonchemical Parameters of Variation

Parameter	Parameter Ranges	
	Min.	Max.
T_{eff} [K]	4000	5000
$\log g$ [dex]	1.5	3.0

real stars. We therefore make only a comparative analysis of different modeling choices in the recovery of abundance twins, rather than a quantitative prediction of performance for real survey data.

4.1. Simulated Data Set

For the creation of our spectra, we relied on the APOGEE package introduced in Bovy (2016), which wraps the Turbospectrum spectral synthesis code (Plez 2012) using ATLAS9 atmospheres (Mészáros et al. 2012). We created identically distributed training and test data sets, both containing 25,000 pairs of chemical abundance twins, sharing identical surface chemical abundances but differing stellar parameters. We generated our spectra assuming solar isotope ratios.

When creating our spectra, the nonchemical parameters varied were the effective temperature T_{eff} and surface gravity $\log g$. For each spectra, T_{eff} and $\log g$ were generated by sampling from uniform distributions, as found in Table 1. These parameter ranges were designed to replicate those of red giant-type stars, which are the favored type of stars for chemical tagging (Hogg et al. 2016; Price-Jones & Bovy 2017). Chemical abundances were generated by independently sampling log-metallicity ([Fe/H]) and log-element abundance enhancements ([X/Fe]), assuming Gaussian-distributed values. Our Gaussian $1 - \sigma$ standard deviations were chosen to roughly reflect those observed by the APOGEE survey. Exact values can be found in Table 2. These were determined from the 1σ element abundance dispersion in APOGEE’s DR14 for each element for red giant stars. By fitting separate 1D Gaussians to the element abundance enhancements, we ignore any further correlations that may exist between elemental abundances. In doing this, we will overestimate the spectral variability (i.e., dimensionality), which could lead to our chemical tagging predictions being overly optimistic, for the set of stars we consider in our tests. The absolute performance that we later report in the recovery of abundance twins is subject to the number of stars we are evaluating, as well as their density in chemical element abundance space and the dimensionality of the spectra themselves. Therefore, it is only the comparative performance between the approaches we show that is relevant.

4.2. Implementation Details

For both FaderDis and FactorDis, we performed a manual hyperparameter search on the training data set to select the best-performing model. Results for selected models are then shown on the test data set. We chose to set the latent dimensionality, z , to have dimension 20, slightly exceeding the number of varied abundances. A more comprehensive description of our neural network architectures can be found in the Appendix. Our code is available on GitHub at <https://github.com/drd13/tagging-package>.

Table 2
Mean and Standard Deviation Used When Sampling the Chemical Factors of Variation

[X/Fe]	Mean (dex)	Standard Deviation (dex)
[Fe/H]	−0.13	0.24
[N/Fe]	0.28	0.11
[O/Fe]	0.03	0.08
[Na/Fe]	−0.05	0.38
[Mg/Fe]	0.06	0.08
[Al/Fe]	0.07	0.09
[Si/Fe]	0.05	0.07
[S/Fe]	0.05	0.07
[K/Fe]	0.04	0.07
[Ca/Fe]	0.02	0.04
[Ti/Fe]	−0.01	0.06
[V/Fe]	−0.01	0.11
[Mn/Fe]	−0.04	0.07
[Ni/Fe]	0.02	0.04
[P/Fe]	−0.04	0.18
[Cr/Fe]	−0.01	0.06
[Co/Fe]	0.	0.15
[Rb/Fe]	−0.03	0.29

Note. Enhancements and metallicity are assumed to be Gaussianly distributed in dex.

We also evaluated the performance of the model developed in Price-Jones & Bovy (2017, 2019; described in Section 2.2), which we refer to as PolyDis from now on. We use PolyDis with a fourth-order polynomial, as was found to work best on a training data set.

To better simulate the real data, we also evaluate our methods on a test data set with added Gaussian noise. For a given S/N, we add to every bin of the continuum-normalized spectrum zero-mean Gaussian noise with a standard deviation $\sigma = \frac{1}{S/N}$. For FactorDis, the results of the noisy test data set are obtained by training models on data in which noise of order 1% (S/N = 100) is added to every observation during training. For FaderDis (and PolyDis), noise of order 1% was added to the training data and kept constant for every epoch of training. It was found that adding this type of noise to the training data set led to worsened spectral reconstruction but improved the isolation of the chemical factors of variation. The worsened reconstruction can easily be attributed to overfitting to the noise. We do not have a clear explanation for why it led to improved isolation of the chemical factors of variation.

5. Results

In this section, we present a series of experiments comparing and contrasting the capacities of the different models.

5.1. Resolving Power of Latent Representation to Distinguish Chemically Identical Stars

If nonchemical factors of variation have been perfectly removed from the latent z , stars sharing a common chemical abundance should, in turn, also share a common latent vector. As such, any difference in latent representation between chemically identical stars can be attributed to imperfections in the learned representation. Here we use this to compare and contrast how well our considered methods isolate out the chemical factors of variation.

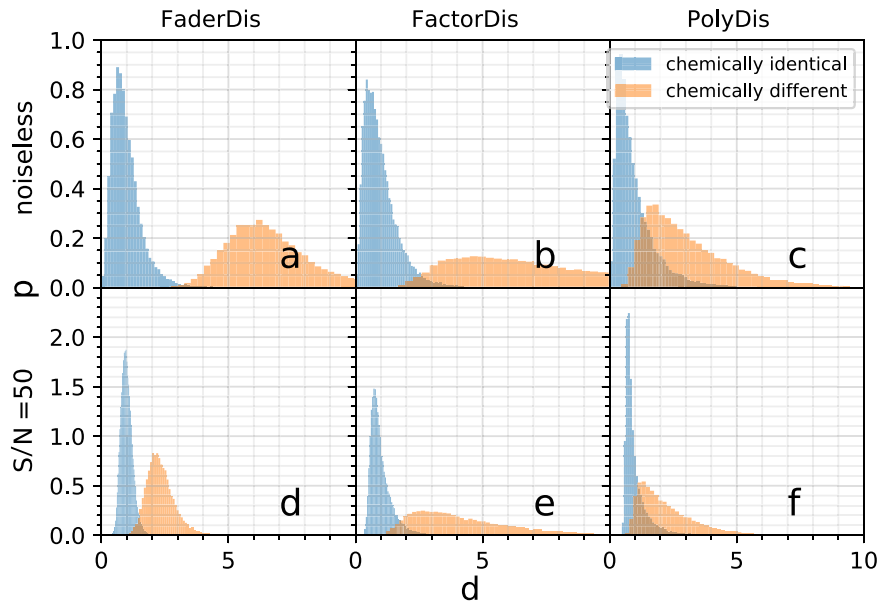


Figure 2. Distribution of scaled Euclidean distances, d , for a sample of chemically identical (blue) and fully randomly sampled (orange) pairs of stars. For each model, a scaling is applied to the latents such that the mean distance of chemically identical stars is 1. Each model includes T_{eff} , $\log g$, and $[\text{Fe}/\text{H}]$ as the parameters to disentangle from the chemical factors of variation. The top row is evaluated using the noiseless test data set and the bottom with noise of order $S/N = 50$ added. The left column is evaluated using the FaderDis method, the middle using the FactorDis method, and right using the PolyDis method (after principal component analysis with 50 components).

Figure 2 shows histograms of Euclidean distances, $\|z_i - z_j\|_2$, calculated on the latents, z , for both chemically identical pairs of stars (blue) and randomly sampled (non-chemically identical) pairs of stars (orange). We show this for our three different disentanglement methods at several S/N s. Distances are evaluated on the 20-dimensional latent for both the FaderDis and FactorDis methods and the 50-dimensional principal component analysis components for the PolyDis approach. We found that 50 principal components explained 99.95% of the variance in the (noiseless) data. In the interest of making the comparison with PolyDis fair, we include $[\text{Fe}/\text{H}]$ as a disentangled parameter during the training of the FaderDis and FactorDis methods.

Reassuringly, we find that for all considered methods, chemically identical stars share more similar latents than random pairs of stars. However, not all methods are equally good at this task, with PolyDis underperforming compared to FaderDis and FactorDis. Indeed, we find that, unlike the other considered methods, the PolyDis method has a nonnegligible overlap between the distributions of chemically identical pairs of stars and random pairs of stars. This means that for chemical tagging purposes, there will be a larger fraction of random stars in the data set falsely appearing more chemically similar than genuinely chemically identical stars.

5.2. Quantifying Chemical Tagging Performance

In this section, we evaluate the quality of our learned representations directly on the task of chemical tagging. Since our data set was designed such that every star has a unique chemical abundance twin, we can evaluate chemical tagging methods based on their capability of recovering these introduced chemical abundance twins. We once again use the Euclidean distance in latent space $d = \|z_i - z_j\|_2$ as our measure of chemical similarity between stars.

We show the results of our analysis in Figure 3, where we have plotted the distribution of “false” chemical abundance

twins recovered with each method—considered to be stars appearing more similar than the genuine chemical abundance twin. We term these our “doppelganger” stars. In our plot, the y -axis corresponds to the percentage of stars in the test data set with fewer false twins than the corresponding value on the x -axis. For example, when evaluating the FactorDis model that was trained to remove $[\text{Fe}/\text{H}]$ on a data set without noise (as shown in panel (d)), we found that around 85% of stars in the data set had fewer than 10 out of the 49,998 other stars in the data set being mistakenly measured as more chemically similar than their genuine chemical abundance twin. Similarly, the y -intercept represents the percentage of stars for which none of the 49,998 other stars in the data set are more similar than the genuine chemical twin.

The figure suggests that precision disentanglement and removal of nonchemical factors of variation from stellar spectra is valuable for chemical tagging pursuits. The FaderDis method identifies significantly more pairs of chemically identical stars than the baseline PolyDis method. For example, the FaderDis method, applied on a noiseless data set with $[\text{Fe}/\text{H}]$ removed from the representation (panel (c)), identifies around 97% of pairs of chemical abundance twins compared to only 50% for the baseline PolyDis method (panel (e)). For spectra with $S/N = 100$, this number goes down to about 88%. As the neural network performance is sensitive to hyperparameters, architecture, and loss function, any improvement in these areas could further improve the results. For example, the FaderDis method was found to perform significantly worse when noise was not added as described in the implementation details.

Note that as we randomly generate our stars from a high-dimensional distribution, there is a possibility for random pairs of stars to be chemically similar by chance. However, we expect a chance of no more than 10^{-12} of doppelganger pairs given the high dimensionality of the artificially generated data set. Even if these exist, however, our figure is comparative only, to demonstrate how the three different methods work to

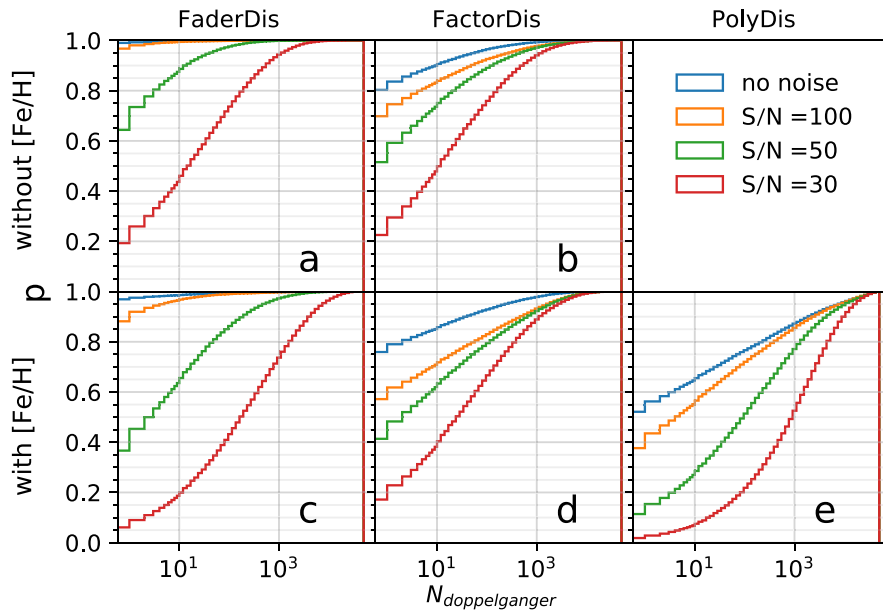


Figure 3. Fraction of false chemical abundance twins for different models with differing S/N. In each panel, we plot the percentage of stars in the test data set with fewer false twins than x , where x is the x -axis value, denoted as $N_{\text{doppelganger}}$. In the top row, we show results conditioned on T_{eff} and $\log g$. In the bottom row, we show results conditioned on T_{eff} , $\log g$, and $[\text{Fe}/\text{H}]$. We plot the results obtained for FaderDis in the left column, with FactorDis in the middle column and PolyDis in the right column. It is worth reemphasizing that $N_{\text{doppelganger}}$ is highly dependent on the size of the data set, and as such, this figure is only intended to be comparative and not an absolute reference.

recover the designated chemical abundance twin stars. As we are generating data from a fixed range of 20 independent abundance labels, recovery of chemical abundance twin stars will become harder under various conditions. This includes the size of our test set growing within its current abundance ranges and if the correlations between the abundances were included in their prescription. We highlight, however, that this is a comparative test to demonstrate the relative performance of the three methods and as a function of S/N. The absolute performance would vary in the physical abundance distribution plane of real data.

5.3. Interpretability of Latent Representation

In this section, we investigate whether the latent representations that organically emerge from our neural networks are interpretable. As our encoder and decoder are nonlinear functions, we might pessimistically expect our latent representations to be noninterpretable. We show that this is not the case and that instead, at least on our synthetic data set, the learned representations align well with the measured abundances.

We approach this question through learning a linear transformation converting from latents to abundances. We represent our data set of abundances and latents as matrices V and Z of shape $n_{\text{species}} \times n_{\text{data}}$ and $n_z \times n_{\text{data}}$, where n_{species} is the number of chemical species in the spectra, n_z is the dimensionality of the latent space, and n_{data} is the number of observations in the data set. We seek a transformation matrix A converting latents into abundances as faithfully as possible. We can find such a matrix by solving $\text{argmin}_A \|AZ - V\|^2$, which has the known solution $A = VZ^+$ (Petersen & Pedersen 2008), with Z^+ the Moore–Penrose inverse of Z . We solve this matrix using all stars in the noiseless training data.

In Figure 4, we have plotted chemical compositions as estimated from the linearly transformed latents against true chemical compositions. These are shown for 2000 stars in the

noiseless test data set. We see a remarkable agreement between the estimated and true abundances. For almost all species, the linear transformation is nearly as good at estimating chemical compositions as a neural network trained on the latents (denoted “nonlinear”) on the same stars. Although Na is not as well fit as other species, it is known to be particularly difficult to estimate (Jönsson et al. 2018; Ness et al. 2019). This shows that our method has naturally learned to decompose spectra into a representation nearly equivalent to chemical abundances. Although these results were obtained on a synthetic data set, they are particularly encouraging. Measuring abundance variation quantitatively without reliance on synthetic spectra would allow for fully circumventing the uncertainties propagated from inaccuracies in spectral modeling.

5.4. Spectral Reconstruction

Our neural network encoder allows for converting spectra into a representation in which predefined nonchemical factors of variation are removed. By subsequently applying the decoder to this representation, we can generate modified spectra recast to new nonchemical parameters.

In Figure 5, we leverage this to visually demonstrate, for the FactorDis approach, how well our learned representation isolates the chemical information in the spectra of a pair of metal-rich stars (top panel) and a pair of metal-poor stars (bottom panel). These test spectra have been generated as described in Section 4.1, with each pair sharing identical chemical compositions but differing physical parameters.

For each panel, in the top plot, we show the original pair of stellar spectra. In the middle plot, we show how these same chemical abundance twins appear after x_1 is transformed to the physical parameter of x_2 , and in the bottom plot, we show the residuals between the twins after the transformation. From these figures, we see that although the initial spectra are very different, the transformed spectra are nearly identical. This is because the encoder isolated the chemical information and the

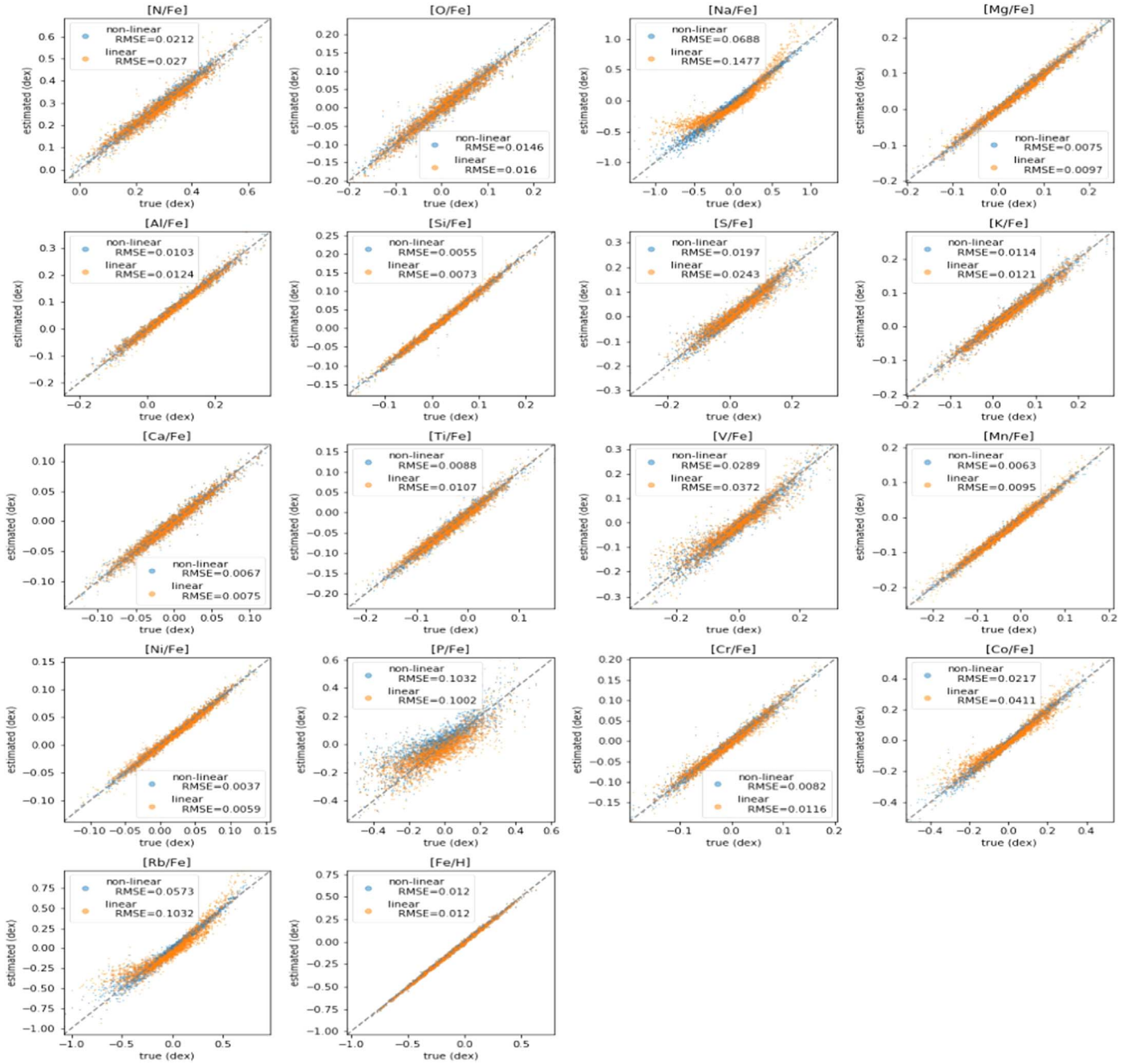


Figure 4. Scatter plot showing estimated against true chemical enhancements and metallicities for synthetic stars in our test data set. In the legend, linear refers to abundances estimated by multiplying the latent with matrix A , and nonlinear refers to abundances estimated from the latent using a neural network. This figure was obtained using the latent from a FaderDis model trained at disentangling $[T_{\text{eff}}, \log g]$. For each chemical element, we have also estimated the rms error (RMSE), the standard deviation of the residuals between predicted and true enhancements/metallicity.

decoder generated the recast spectra (for star x_1) at the newly provided physical parameters (of the star x_2).

In Figure 6, we show the residuals between a star and its transformed twin for FactorDis, FaderDis, and PolyDis. For this comparison, we include the three factors of nonchemical variation, T_{eff} , $\log g$, and $[\text{Fe}/\text{H}]$, in the disentanglement network training. Alongside the figure, we also report the mean absolute residual across the full spectral region considered, as well as the standard deviation (per pixel) of the residuals, σ_R , for each approach. For the PolyDis method, a star is recast to its chemical abundance twin star’s stellar parameters by replacing its residuals from the polynomial fit with those of its twin star

(the fit is meant to isolate the chemical information in the residuals).

In Table 3, we report the average mean absolute residual $\langle R \rangle$ and average mean squared error $\langle \text{MSE} \rangle$, obtained by averaging over random pairs of chemically identical stars in the data set and transformed to each other’s physical parameters. The $\langle \text{MSE} \rangle$ metric more severely penalizes large deviations in the reconstructed compared to the original spectra. Several interesting trends appear in the data.

We observe a difference in performance between methods, depending on whether the residuals or the squared residuals are used for evaluation. Most notably, the FactorDis method

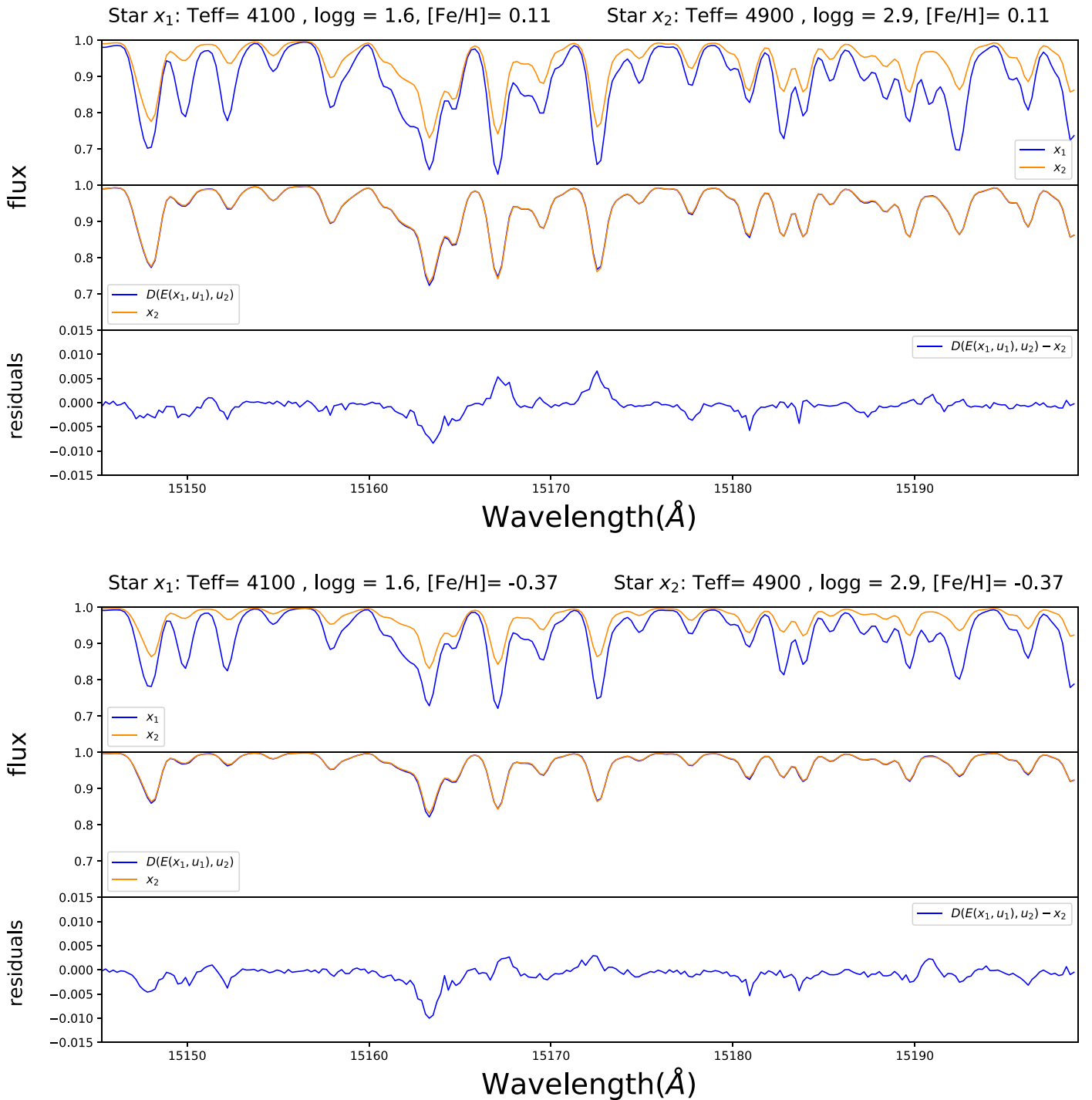


Figure 5. For each panel, in the top plot, we show the spectra of two stellar chemical abundance twins (with differing T_{eff} and $\log g$), x_1 and x_2 . In the middle plot, the spectra of the second chemical abundance twin, x_2 , is shown with a spectra reconstructed by the decoder ($D(E(x_1, u_1), u_2)$) using the other star’s latent z_1 but the same physical parameters u_2 . In the bottom plot, we show the corresponding residuals. The stellar parameters are shown above each panel (for conciseness, the $[X/\text{Fe}]$ vector is not shown). We can see that the spectra of chemical abundance twins are nearly indistinguishable after transforming them to a common physical parameter (T_{eff} and $\log g$) parameterization.

outperforms the PolyDis in terms of squared but not raw residuals. As squared values are more sensitive to outliers, this seems suggestive that the PolyDis method has comparatively better overall reconstruction but struggles with representing some portions of the data set.

The relative mean values reported in Table 3 are also largely indicative of the distribution in the parameters of our library of test spectra that we have generated. Simpler modeling

approaches likely perform very well when both the training and test data are smaller in overall variability and for pairs of stars that have nearer T_{eff} and $\log g$ parameters. In this case, the spectral variability due to the parameter and abundance labels is nearer to a linear or low-order polynomial form (e.g., Ness et al. 2015; Casey et al. 2016). In the regime where pairs of stars have large differences in their T_{eff} or $\log g$, the move to more complex models (or as an alternative, local linear models

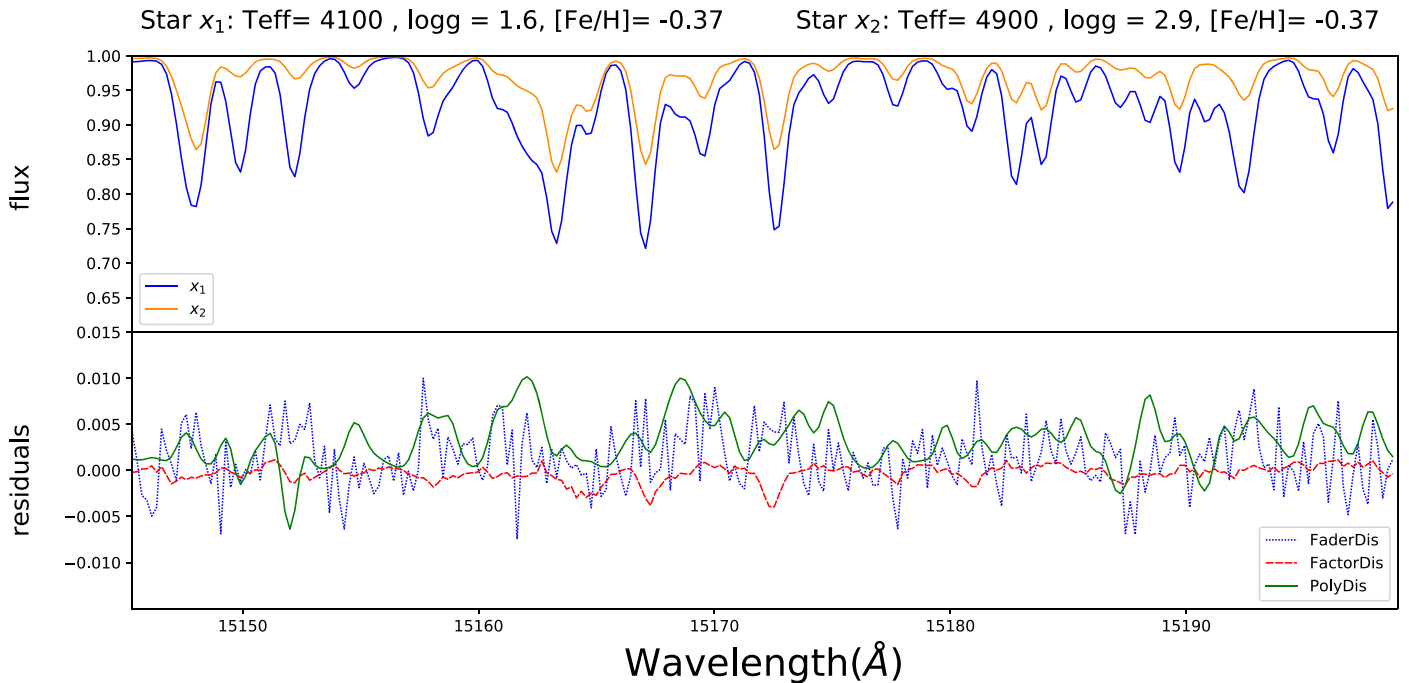


Figure 6. This figure compares the reconstruction capacities of the three disentanglement methods for the metal-rich star shown in Figure 2. In the top panel, we show the spectra of two chemical abundance twins, x_1 and x_2 , for the first 256 wavelength bins. In the bottom panel, we show the residuals between the second twin, x_2 , alongside the spectra of the first twin, x_1 , recast by the decoder ($D(E(x_1, u_1), u_2)$) to the physical parameters u_2 for the three disentanglement methods considered. The mean residuals and associated standard deviation (per pixel across the full spectral range) are $R = 0.0029$ and $\sigma_R = 0.0021$ for FaderDis, $R = 0.0011$ and $\sigma_R = 0.0009$ for FactorDis, and $R = 0.0034$ and $\sigma_R = 0.0023$ for PolyDis.

Table 3

Average MSE between Two Chemically Identical Stars Transformed to Each Other’s Physical Parameters for the Different Methods

Method	$\langle R \rangle$	$\langle \text{MSE} \rangle$
FactorDis	0.0021	1.26×10^{-5}
FaderDis	0.0030	1.68×10^{-5}
PolyDis	0.0018	1.50×10^{-5}

Note. The quoted number assumes a data set of stars distributed following the procedure as described in Section 4.1.

that build nonparametric models using nearest neighbors (e.g., Wheeler et al. 2021), may have higher return. These differences can also be understood in terms of the differences between methods at reconstructing spectra. In the PolyDis method, spectra are recast to new physical parameters by adding residuals to the polynomial fit. This transformation is not parametric in the traditional sense; so, if two stars being compared are similar to begin with, they will give a very small reconstruction loss. For the case of pairs of identical spectra, this would give a perfect reconstruction, even if the residuals do not capture the chemical information. On the other hand, the FactorDis and FaderDis methods involve decoding from a lower dimensional representation; so, even identical stars will have nonzero residuals. The difference thus boils down to FactorDis and FaderDis being, by design, built for capturing chemical information but not always (for our exercise) reconstructing the stellar spectra, while the PolyDis method can “cheat” at reconstructing stars. The FaderDis method does not perform particularly well at this task. We believe this to be linked to its training procedure, which involves reconstructing noisy rather than clean data. To demonstrate that our method performs better on chemically dissimilar stars, we have

Table 4

Average Reconstruction between Two Chemically Identical Stars Transformed to Each Other’s Physical Parameters for the Different Methods on a Restricted Data Set Composed of Stellar Chemical Abundance Twin Pairs with at Least 500 K of Temperature Difference

Method	$\langle R \rangle$	$\langle \text{MSE} \rangle$
FactorDis	0.0027	2.13×10^{-5}
FaderDis	0.0033	2.14×10^{-5}
PolyDis	0.0029	3.33×10^{-5}

recalculated our metrics on a data set restricted to stars with high-temperature differences (see Table 4). On this partial data set, the FactorDis method performs better across the board, and the PolyDis method performs worse than both FaderDis and FactorDis, in terms of $\langle \text{MSE} \rangle$.

6. Discussion

We have developed a neural network architecture to remove those factors of variation in stellar spectra that we want to disregard from those that we care about. Here we want to isolate the chemical abundances. Typically, chemical abundances are measured from stellar spectra, which relies on imperfect stellar models and does not fully utilize the full amount of information across the entire spectral region. We seek to develop approaches that circumvent limitations in our current knowledge of stellar physics or incomplete models and leverage large surveys by working as close to the observed data space as possible.

We compared two deep-learning approaches and a simpler polynomial model approximation for the task of removing T_{eff} , $\log g$, and $[\text{Fe}/\text{H}]$ from model mock stellar spectra, leaving behind the intrinsic variation caused by chemical abundances.

All three approaches perform well at generating a disentangled representation of spectra. A reader might note that the mean residual of the test data compared to the model, for all three methods we investigate, is lower than a typical Poisson noise level in observed spectra in large surveys (often $S/N \approx >50$ – 100 per wavelength). The value of this level of precision and the importance of maximizing the precision comes when working with large stellar ensembles. Galactic archeology typically demands large numbers of stars where the sampling precision of the population increases as $(N)^{1/2}$. The sampling error of the population itself at each wavelength becomes smaller than the reconstruction precision for very large surveys.

We demonstrated the benefits of using a disentangled neural network with a chemical abundance twin star recovery from our 50,000 star test set. This is related to the pursuit of chemical tagging, which is an extremely challenging aspiration with galactic archeology—to find stars born together using their identical abundances. Even if chemical tagging is prohibited by field contamination, there is tremendous promise in modeling the distribution of the most chemically similar stars as a function of orbital properties or abundance density in reconstructing galactic history (e.g., Kamdar et al. 2019; Coronado et al. 2020; Price-Jones et al. 2020). When applying our FaderDis approach to a synthetic data set of APOGEE-like stars, we were able to identify the most chemically identical stars from the ensemble, even with a moderate S/N . At an S/N of 100, we were able to identify around 87% of the pairs of stars. This compared to around 60% of the stars for our second neural network approach, FactorDis, and around 40% for our implementation of a polynomial representation to subtract stellar parameters, PolyDis, at this S/N .

Our results obtained using synthetic model spectra are particularly promising. However, these may not translate directly to real survey data for a number of reasons. As our data set was handcrafted, we were able to ensure that it perfectly matched the stringent requirements of our method. That is to say, we ensured perfect knowledge of all nonchemical factors of variation and expressed these using a deterministic parameterization that is statistically independent from the chemical factors of variation. We examine here whether these assumptions are accurate for actual stellar surveys, and if not, how we might be able to modify our method to accommodate these discrepancies.

6.1. Assumptions about Stellar Spectra

Our method involves removing all nonchemical factors of variation. If our neural network is conditioned on an incomplete set of nonchemical factors of variation u , our latent will be contaminated by these when isolating chemical factors of variation. For actual observations, these nuisance parameters may arise from imperfect calibration, such as from telluric lines or persistence in the detector or any other of a number of systematics. In principle, we may be able to somewhat counteract this phenomenon by restricting the dimensionality of our latent. This would force the latent to only encode the most important factors of variation. However, as some abundances only have a minuscule impact on the overall recorded spectral flux, we require very good knowledge of our factors of variation. An alternative approach for accounting for these systematics would be to add a disentanglement term targeting them. However, this would require additional

infrastructure not built into this first demonstration of the approach.

In our proof-of-concept experiments, we first modeled stars using only the effective temperature T_{eff} and surface gravity $\log g$ as nonchemical factors of variation. These two parameters should explain most of the nonchemical variance in the spectral data. Indeed, many data-driven models have been capable of accurately reconstructing spectra using these parameters, plus overall metallicity $[\text{Fe}/\text{H}]$ (i.e., Leung & Bovy 2018), as these are responsible for the majority of spectral variability. However, other parameters that are independent of the chemical composition may also impact the observed spectra and so may need to be included in our conditioning parameters u . Stellar mass or age, for example, while correlated with effective temperature and surface gravity (Price-Jones & Bovy 2017), contains additional independent predictive power for generating the spectra (Ness et al. 2016). If this is indeed the case, it may be beneficial to include an independent estimate of stellar mass. This could be achieved by using a training data set of stars from asteroseismology surveys with mass estimates. Stellar rotation may also affect stellar spectra. These variations may, at least in part, be captured by the micro- and macroturbulence parameters.

Beyond assuming knowledge of nonchemical factors of variation, we have also so far assumed the ability to perfectly estimate these, if known. In realistic scenarios, this may not be easy. However, similarly to other data-driven methods, such as Ness et al. (2016), our method requires precise but not necessarily accurate parameter values. For example, our neural network method should still be effective if a change of variable is applied to any of the conditioning variables. Furthermore, we do not account for the correlations between elements when we generate our test data, which will reduce the dimensionality of the spectra and effective sparsity of the data space.

Finally, even if we are unable to fully remove nonchemical factors of variation from spectra, our neural network architecture may still be useful for traditional chemical abundance estimation. Indeed, we may be able to reduce systematic uncertainties in traditional abundance estimation methods by recasting stars to a common temperature and surface gravity (as shown in Section 5.4) before comparing to synthetic stellar spectra. Similarly to differential analysis, this would serve to restrict the number of factors of variation changing in stellar spectra.

6.2. Assumptions Relating to Statistical Independence

Our approach assumes that abundances are statistically independent from other factors of variation. In our experiments, the synthetic spectral data set was generated so as to satisfy this assumption. This assumption is not entirely unreasonable. There is evidence that most stellar abundances should, at least to first order, be independent from temperatures and surface gravity (Jofré et al. 2019). Trends between abundances and physical parameters have, in the past, been attributed to systematic uncertainties and sometimes even been corrected for (Valenti & Fischer 2005; Adibekyan et al. 2012). However, overall metallicity does, at some level, affect stellar evolution (e.g., see Gaia Collaboration et al. 2018a). Ultimately, this assumption breaks down to some degree, and there is some level of statistical dependency between metallicity and physical parameters in observed spectra. Including overall metallicity in the disentangled parameters, as in our experiments, mitigates

this issue. Spectral synthesis approaches, including at low resolution, typically derive a basic set of T_{eff} , $\log g$, and $[M/H]$ (or $[Fe/H]$) parameters (with errors), so all parameters, including metallicity, are readily available to use in the disentanglement architecture we have built. We might also identify chemically identical stars by finding those stars sharing both a common latent representation and a common metallicity. Ultimately, accounting for any dependencies between abundances and parameters will better disentangle the abundance variations from stellar parameter variations in real spectra. Large data sets may be leveraged to learn these dependencies. Indeed, stellar processes like dredge-up and diffusion (Mason & Gilmore 2015; Martig et al. 2016) modify surface abundances away from their birth values across evolutionary states. Removing any trends caused by these processes would result in a chemical representation closer to birth abundances, which is ultimately preferable for using abundances for chemical tagging pursuits.

6.3. Beyond Synthetic Spectra

There are a few challenges associated with applying our method to real observations. Spectral bins in real observations are sometimes flagged as untrustworthy, for example, due to cosmic rays or persistence in the detector (see Jönsson et al. 2020). These are flagged for each APOGEE spectrum, and individual pixels are correspondingly masked. Since our neural network methodology requires all spectral bins as inputs, such untrustworthy or missing data need to be imputed somehow, so as to not impact the downstream learned representation. Another more practical challenge with applying the method to real observations is that it requires hyperparameter tuning, and training the algorithm takes a day to run on specialized hardware (GPUs). This makes iterative deployment slow.

7. Conclusion

Organizing stars by their chemical similarity and investigating the distribution of their other properties (e.g., orbits, density) is a promising avenue for unraveling galactic evolution (i.e., Ting et al. 2016; Kamdar et al. 2019; Coronado et al. 2020). Ranking stars by chemical similarity requires precise chemical information for large numbers of stars. Chemical similarity is typically determined using measured element abundances. However, these measurements are subject to inaccuracies and systematics that are inherited from incomplete and approximate stellar models. As an alternative to deriving abundances from spectra, using the variability of the spectra themselves becomes possible and advantageous in the regime of large stellar surveys. Data-driven deep-learning methods—applied directly to the spectra themselves—find natural applications here.

In this paper, we introduce a new deep-learning method for extracting chemical information from spectra. This relies on isolating chemical factors of variation from nonchemical factors of variation through training a neural network using a disentanglement loss. This method removes the need for accurate and precise modeling of the chemical abundance factors of variation in stellar spectra. Instead, it relies only on the parameterization of the other primary sources of variability, namely stellar parameters, including T_{eff} and $\log g$. This requires conditioning a neural network on these factors, in our case, T_{eff} and $\log g$ (and also $[Fe/H]$ for modeling only the

variation from abundance enhancements, $[X/Fe]$). We have shown, using a synthetic set of spectra that we have generated, that our method can be used to accurately identify and distinguish chemically identical pairs of stars from a field distribution, which is an aim of chemical tagging. We were able to identify more than 85% of the pairs of chemical abundance twins from a data set of 50,000 spectra generated at the resolution of APOGEE and an S/N of 100 with 20 independently drawn chemical abundances ($[X/Fe]$). To do this in practice (on real data) will require being able to estimate all nonchemical factors of variation in the spectra very well. In our analysis, as we wanted to demonstrate our method on a toy data set with the fewest assumptions possible, we have treated the metallicity as statistically independent from physical parameters. For real observations, however, it may be beneficial to account for their statistical dependency. As chemical and physical parameters are believed to be independent, or close to it, at fixed metallicity, we suggest learning a representation in which metallicity is disentangled.

We note that our approach may also find utility in regimes where analysis is hindered by a large number of molecular bands in the spectra with uncertain atomic transition data that render stellar models inaccurate. Our tests here are confined to a very narrow range of T_{eff} that is not dominated by molecular features, but this may be a promising avenue for large survey data like Sloan V that will observe bright, cool giants with molecular features (Kollmeier et al. 2017).

In this paper, through experiments on a synthetic data set, we have demonstrated the efficacy of our newly proposed method for extracting chemical information from stellar spectra. These experiments act as a proof of concept in a controlled environment, where the data-generating process is perfectly understood. This sets out the groundwork for applying such representation learning methods to real observations. The natural next steps of this line of work will be translating the success found on synthetic spectra to real APOGEE observations. Because of the imprint of systematics on real stellar spectra and other possible departures from our assumptions, this will likely require further modifications and/or fine-tuning of the approach. Such investigations are reserved for a future paper. In a forthcoming paper (D. de Mijolla et al. 2021, in preparation), we demonstrate on real APOGEE stellar spectra a method similar but complementary to this one for extracting chemical information in a manner that is robust to instrumental systematics.

Beyond our astronomical contributions, we hope that our proposed methodology will find uses in other fields. Our application of supervised disentanglement for identifying observations sharing a common parameterization is a novel method that could be adapted to other tasks. In particular, our chemical tagging experiments and associated data sets could be useful in comparing different supervised disentanglement architectures, something that has so far been lacking in the machine-learning community. We believe that our task of evaluating how well a supervised disentanglement neural network maps chemically identical stars to an identical latent is particularly useful for assessing supervised disentanglement. This is because it does not rely on training any secondary networks and gives a single value that is directly indicative of the level of disentanglement found in the latent. Our proposed novel supervised disentanglement architecture has shown good performance at chemical tagging-like pursuits and

disentanglement that suggests that it may be a competitive alternative to Fader disentanglement types of architectures.

D.D.M. is supported by the STFC UCL Centre for Doctoral Training in Data Intensive Science (grant No. ST/P006736/1). The authors also acknowledge the Flatiron Institute for use of their HPC system, which allowed this work to be performed. The authors would also like to thank Ioanna Manolopoulou for helpful discussions, Jo Bovy for maintaining the APOGEE python package, and David W. Hogg for helping connect the coauthors.

Appendix Neural Network Training Details

We briefly review some implementation details useful for reproducing the results in this paper. We have made our repository open-source to aid in making our paper reproducible and encourage readers to refer to the code for additional details.

Data set processing. We process the continuum-normalized spectra by first multiplying the spectra by 4 and then subtracting 3.5. This makes the spectra roughly occupy the $[-1, 1]$ range.

Neural network training. All quoted results use feed-forward neural networks with self-normalized rectified unit activation functions (Klambauer et al. 2017). All results are obtained using the ADAM optimizer (Kingma & Ba 2015) with a learning rate of 10^{-5} . In the following, $n_{\text{bins}} = 7751$ refers to the number of spectral bins used, $n_{\text{conditioned}} = 2$ or 3 is the number of parameters the encoder is conditioned on, and $n_z = 20$ is the size of the autoencoder latent.

FactorDis architecture. Our FactorDis neural network has the following architecture (including input and output layers). Results can be reproduced with the loss-weighting term $\lambda = 10^{-4}$:

$$\begin{aligned} &\text{encoder dimensions} \\ &= \{n_{\text{bins}} + n_{\text{conditioned}}, 2048, 512, 128, 32, n_z\}, \end{aligned} \quad (\text{A1})$$

$$\begin{aligned} &\text{decoder dimensions} \\ &= \{n_z + n_{\text{conditioned}}, 512, 2048, 8192, n_{\text{bins}}\}, \end{aligned} \quad (\text{A2})$$

$$\begin{aligned} &\text{discriminator dimensions} \\ &= \{n_{\text{bins}} + n_{\text{conditioned}} + n_z, 4096, 1024, 512, 128, 32, 1\}. \end{aligned} \quad (\text{A3})$$

FaderDis architecture. Our FaderDis neural network has the following architecture (including input and output layers). Results can be reproduced with the loss-weighting term $\lambda = 10^{-5}$. When training the auxiliary network, each disentangled parameter was split into 10 discrete values, creating 100 equal-sized bins when disentangling two parameters and 1000 equal-sized bins when disentangling three:

$$\begin{aligned} &\text{encoder dimensions} \\ &= \{n_{\text{bins}} + n_{\text{conditioned}}, 2048, 512, 128, 32, n_z\}, \end{aligned} \quad (\text{A4})$$

$$\begin{aligned} &\text{decoder dimensions} \\ &= \{n_z + n_{\text{conditioned}}, 512, 2048, 8192, n_{\text{bins}}\}, \end{aligned} \quad (\text{A5})$$





$$\begin{aligned} &\text{auxiliary dimensions} \\ &= \{n_z + n_{\text{conditioned}}, 512, 256, 10^{n_{\text{conditioned}}}\}. \end{aligned} \quad (\text{A6})$$

Nonlinear chemical estimation. In Figure 4, neural networks, taking latents z as inputs, are used as nonlinear estimators of abundances. A separate neural network with the following

structure was trained for every chemical species:

$$\text{nonlinear dimensions} = \{n_z, 512, 256, 128, 1\}. \quad (\text{A7})$$

ORCID iDs

Damien de Mijolla  <https://orcid.org/0000-0001-8757-4936>
Melissa Kay Ness  <https://orcid.org/0000-0001-5082-6693>
Serena Viti  <https://orcid.org/0000-0001-8504-8844>
Adam Joseph Wheeler  <https://orcid.org/0000-0001-7339-5136>

References

- Adibekyan, V. Z., Sousa, S. G., Santos, N. C., et al. 2012, *A&A*, 545, A32
Arjovsky, M., Chintala, S., & Bottou, L. 2017, ICML, 34, 214, <http://proceedings.mlr.press/v70/arjovsky17a.html>
Beane, A., Ness, M. K., & Bedell, M. 2018, *ApJ*, 867, 31
Bedell, M., Bean, J. L., Meléndez, J., et al. 2018, *ApJ*, 865, 68
Belghazi, M. I., Baratin, A., Rajeswar, S., et al. 2018, ICML, 35, 531, <http://proceedings.mlr.press/v80/belghazi18a.html>
Bengio, Y., Courville, A. C., & Vincent, P. 2013, *ITPAM*, 35, 1798
Bertelli Motta, C., Pasquali, A., Richer, J., et al. 2018, *MNRAS*, 478, 425
Blanco-Cuaresma, S., & Fraix-Burnet, D. 2018, *A&A*, 618, A65
Bonifacio, P., Dalton, G., Trager, S., et al. 2016, in Proc. Annual Meeting of the French Society of Astronomy I & Astrophysics Lyon, ed. C. Reylé (Paris: Société Française d’Astronomie et d’Astrophysique - SF2A), 267
Bovy, J. 2016, *ApJ*, 817, 49
Casali, G., Spina, L., Magrini, L., et al. 2020, *A&A*, 639, A127
Casey, A. R., Hawkins, K., Hogg, D. W., et al. 2017, *ApJ*, 840, 59
Casey, A. R., Ho, A. Y. Q., Ness, M., et al. 2019, *ApJ*, 880, 125
Casey, A. R., Hogg, D. W., Ness, M., et al. 2016, arXiv:1603.03040
Chen, J., Konrad, J., & Ishwar, P. 2019, arXiv:1906.09313
Chen, X., Duan, Y., Houthoofd, R., et al. 2016, in 30th Conference on Neural Information Processing Systems (Barcelona) 2172, <https://proceedings.neurips.cc/paper/2016/hash/7c9d0b1f96aebd7b5eca8c3edaa19ebb-Abstract.html>
Coronado, J., Rix, H.-W., Trick, W. H., et al. 2020, *MNRAS*, 495, 4098
Cui, X.-Q., Zhao, Y.-H., Chu, Y.-Q., et al. 2012, *RAA*, 12, 1197
de Jong, R. S., Barden, S. C., Bellido-Tirado, O., et al. 2016, *Proc. SPIE*, 9908, 99081O
De Silva, G. M., Freeman, K. C., Bland-Hawthorn, J., et al. 2015, *MNRAS*, 449, 2604
Dotter, A., Conroy, C., Cargile, P., & Asplund, M. 2017, *ApJ*, 840, 99
Edwards, H., & Storkey, A. J. 2016, ICLR, arXiv:1511.05897
Feng, Y., & Krumholz, M. R. 2014, *Natur*, 513, 523
Feuillet, D. K., Frankel, N., Lind, K., et al. 2019, *MNRAS*, 489, 1742
Frankel, N., Rix, H.-W., Ting, Y.-S., Ness, M., & Hogg, D. W. 2018, *ApJ*, 865, 96
Freeman, K., & Bland-Hawthorn, J. 2002, *ARA&A*, 40, 487
Gaia Collaboration, Babusiaux, C., van Leeuwen, F., et al. 2018a, *A&A*, 616, A10
Gaia Collaboration, Brown, A. G. A., Vallenari, A., et al. 2018b, *A&A*, 616, A1
Ganin, Y., Ustinova, E., Ajakan, H., et al. 2016, *J. Mach. Learn. Res.*, 17, 2096, <http://dl.acm.org/citation.cfm?id=2946645.2946704>
Gilmore, G., Randich, S., Asplund, M., et al. 2012, *Msngr*, 147, 25
Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al. 2014, in Advances in Neural Information Processing Systems 27 (NIPS 2014) (Montreal) 2672, <https://papers.nips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html>
Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. 2017, in Advances in Neural Information Processing Systems 30: Annual Conf. on Neural Information Processing Systems (Long Beach, CA) 5767, <https://proceedings.neurips.cc/paper/2017/hash/892c3b1c6dcd52936e27cbd0ff683d6-Abstract.html>
Hadam, N., Wolf, L., & Shahar, M. 2018, in IEEE/CVF Conf. on Computer Vision and Pattern Recognition (Piscataway, NJ: IEEE), 772
Hawkins, K., & Wyse, R. F. G. 2018, *MNRAS*, 481, 1028
Higgins, I., Matthey, L., Pal, A., et al. 2017, in ICLR <https://openreview.net/forum?id=Sy2fzU9gl>
Ho, A. Y. Q., Ness, M. K., Hogg, D. W., et al. 2017, *ApJ*, 836, 5
Hogg, D. W., Casey, A. R., Ness, M., et al. 2016, *ApJ*, 833, 262

- Holtzman, J. A., Shetrone, M., Johnson, J. A., et al. 2015, *AJ*, **150**, 148
- Jha, A. H., Anand, S., Singh, M., & Veeravasarapu, V. S. R. 2018, in *Computer Vision – ECCV 2018. ECCV 2018, Lecture Notes in Computer Science*, Vol. 11207, ed. V. Ferrari et al. (Cham: Springer), 829
- Jofré, P., Das, P., Bertranpetit, J., & Foley, R. 2017, *MNRAS*, **467**, 1140
- Jofré, P., Heiter, U., & Soubiran, C. 2019, *ARA&A*, **57**, 571
- Jönsson, H., Holtzman, J. A., Prieto, C. A., et al. 2020, *AJ*, **160**, 120
- Jönsson, H., Prieto, C. A., Holtzman, J. A., et al. 2018, *AJ*, **156**, 126
- Kamdar, H., Conroy, C., Ting, Y.-S., et al. 2019, *ApJ*, **884**, 173
- Kingma, D. P., & Ba, J. 2015, in 3rd Int. Conf. for Learning Representations (San Diego, CA)
- Klambauer, G., Unterthiner, T., Mayr, A., & Hochreiter, S. 2017, in *Advances in Neural Information Processing Systems 30: Annual Conf. on Neural Information Processing System* (Long Beach, CA) 971, <https://proceedings.neurips.cc/paper/2017/hash/5d44ee6f2c3f71b73125876103c8f6c4-Abstract.html>
- Kollmeier, J. A., Zasowski, G., Rix, H.-W., et al. 2017, arXiv:1711.03234
- Krumholz, M. R., McKee, C. F., & Bland-Hawthorn, J. 2019, *ARA&A*, **57**, 227
- Lample, G., Zeghidour, N., Usunier, N., et al. 2017, in *Advances in Neural Information Processing Systems 30: Annual Conf. on Neural Information Processing Systems* (Long Beach, CA) 5969, <https://proceedings.neurips.cc/paper/2017/hash/3fd60983292458bf7dee75f12d5e9e05-Abstract.html>
- Leung, H. W., & Bovy, J. 2018, *MNRAS*, **483**, 3255
- Lezama, J. 2019, in ICLR <https://openreview.net/forum?id=Hkg4W2AcFm>
- Liu, F., Asplund, M., Yong, D., et al. 2019, *A&A*, **627**, A117
- Lloyd, S. P. 1982, *ITIT*, **28**, 129
- Locatello, F., Bauer, S., Lucic, M., et al. 2019, in ICLR <https://openreview.net/forum?id=Byg6VhUp8V>
- Louizos, C., Swersky, K., Li, Y., Welling, M., & Zemel, R. S. 2016, ICLR, arXiv:1511.00830
- Majewski, S. R., Schiavon, R. P., Frinchaboy, P. M., et al. 2017, *AJ*, **154**, 94
- Martell, S. L., Shetrone, M. D., Lucatello, S., et al. 2016, *ApJ*, **825**, 146
- Martig, M., Fouesneau, M., Rix, H.-W., et al. 2016, *MNRAS*, **456**, 3655
- Masseron, T., & Gilmore, G. 2015, *MNRAS*, **453**, 1855
- Mészáros, S., Allende Prieto, C., Edvardsson, B., et al. 2012, *AJ*, **144**, 120
- Ness, M., Hogg, D. W., Rix, H.-W., Ho, A. Y. Q., & Zasowski, G. 2015, *ApJ*, **808**, 16
- Ness, M., Hogg, D. W., Rix, H. W., et al. 2016, *ApJ*, **823**, 114
- Ness, M., Rix, H.-W., Hogg, D. W., et al. 2018, *ApJ*, **853**, 198
- Ness, M. K., Johnston, K. V., Blancato, K., et al. 2019, *ApJ*, **883**, 177
- O’Brian, T., Ting, Y.-S., Fabbro, S., et al. 2021, *ApJ*, **906**, 130
- Petersen, K. B., & Pedersen, M. S. 2008, *The Matrix Cookbook* (Copenhagen: Technical Univ. Denmark) <http://www2.imm.dtu.dk/pubdb/p.php?3274>
- Plez, B. 2012, *Turbospectrum: Code for Spectral Synthesis*, Astrophysics Source Code Library, ascl:1205.004
- Polykovskiy, D., Zhebrak, A., Vetrov, D., et al. 2018, *Molecular Pharmaceutics*, **15**, 4398
- Price-Jones, N., & Bovy, J. 2017, *MNRAS*, **475**, 1410
- Price-Jones, N., & Bovy, J. 2019, *MNRAS*, **487**, 871
- Price-Jones, N., Bovy, J., Webb, J. J., et al. 2020, *MNRAS*, **496**, 5101
- Randich, S., Gilmore, G. & Gaia-ESO Consortium 2013, *Msngr*, **154**, 47
- Schiavon, R. P., Johnson, J. A., Frinchaboy, P. M., et al. 2017, *MNRAS*, **466**, 1010
- Schmidhuber, J. 1991, *Learning Factorial Codes By Predictability Minimization*, Tech. Rep. CU-CS-565-91, Univ. Colorado Boulder 565
- Simpson, J. D., Martell, S. L., Da Costa, G., et al. 2019, *MNRAS*, **482**, 5302
- Souto, D., Prieto, C. A., Cunha, K., et al. 2019, *ApJ*, **874**, 97
- Steinmetz, M., Zwitter, T., Siebert, A., et al. 2006, *AJ*, **132**, 1645
- Tamura, N., Takato, N., Shimono, A., et al. 2016, *Proc. SPIE*, **9908**, 99081M
- Ting, Y.-S., Conroy, C., & Rix, H.-W. 2016, *ApJ*, **816**, 10
- Ting, Y.-S., Conroy, C., Rix, H.-W., & Cargile, P. 2019, *ApJ*, **879**, 69
- Ting, Y.-S., Freeman, K. C., Kobayashi, C., De Silva, G. M., & Bland-Hawthorn, J. 2012, *MNRAS*, **421**, 1231
- Valenti, J. A., & Fischer, D. A. 2005, *ApJS*, **159**, 141
- Weinberg, D. H., Holtzman, J. A., Hasselquist, S., et al. 2019, *ApJ*, **874**, 102
- Wheeler, A., Ness, M., Buder, S., et al. 2020, *ApJ*, **898**, 58
- Wheeler, A. J., Hogg, D. W., & Ness, M. 2021, *ApJ*, **908**, 247