# Towards a Spectrograph Digital Twin: Forward Modeling Pixels of 2D Échellograms with Interpretable Machine Learning

Michael Gully-Santiago[1] and TBD[2]

[1] *University of Texas at Austin Department of Astronomy*
[2] *TBD*

## ABSTRACT

We introduce autodiff for 2D echellogram forward modeling.

*Keywords:* data: data analysis — stars: statistics

## 1. INTRODUCTION

Virtually all modern insights derived from observational optical and near-infrared spectroscopy first arrive through the lossy, imperfect, and unavoidable process of digitization onto a 2-dimensional (2D) focal plane array detector. Moore's law has propelled the amount of spectrosopic information we can cram onto these detectors over the last 50 years, with the number of pixels catapulting up to a billion-fold. Innovative optical designs have made evermore effective use of these pixels with information-rich multi-object, imager slicer, and èchellogram spectral formats. The computerized distillation of these spectrograms is generally turnkey. Facility reduction pipelines provide the humdrum "extraction" of raw 2D pixels to familiar 1D extracted spectra, ideally an afterthought in a practioner's journey from observation to scientific results.

But increasingly our greatest scientific ambitions are pushing against the limits of how much information we can squeeze out of a given spectrographic observation. The discovery of other Earths, the redshift of the epoch of reionization, and some of the most important unrealized astronomical discoveries in a generation will occur in the margins of what our most powerful spectrographs can deliver.

The measurement of Extreme Precision Radial Velocity (EPRV) stands out as particularly susceptible to the vagueries of the spectral extraction process. The instrumental Radial Velocity (RV) precision needed to detect an Earth-like planet around a Sun-like star amounts to < 10 cm/s, equivalent to a distance of 10 silicon atoms in a typical spectrograph pixel (Mahadevan et al. 2014). Sub-pixel flat field variations (Vorobiev et al. 2018), charge transfer inefficiencies (Ahmed et al. 2020), and other seemingly pathological materials science phenomena matter at these precisions. Spectral extraction is *not* turnkey at the level of 10 Si atoms. These precision demands have catalyzed a renewed interest in understanding the delicate interplay of the incident stellar spectrum with the detector pixels.

Most 2D échellogram data reduction pipelines can be described as data-driven extraction procedures as opposed to interpretable scene models. The most auspicious of these procedures may be "Optimal Extraction" (Horne 1986), which treats each column as an independent example of a 1D line profile, enabling a judicious weighting in the cross-dispersion direction. This prescription faithfully incorporates low-but-significant Signal-to-Noise Ratio (SNR) pixels. Translating Optimal Extraction to échelle spectra gets complicated by the dense and sometime overlapping layouts of the so-called échellogram format. Spectral traces generally do not align with the pixel coordinates, often curving into smile-shaped arcs. Specialized methods have been developed to handle such layouts (Piskunov & Valenti 2002; Zechmeister et al. 2014; Piskunov et al. 2020).

The desire to reach the elusive photon-noise limit motivated a subtle mental leap compared to Optimal Extraction. *Spectroperfectionism* (Bolton & Schlegel 2010) represents a *scene model*, allowing 2D PSF shapes with possibly asymmetric morphologies, convolved with an underlying super-resolution 1D spectrum with unknown spectral structure. This 2D PSF better resembles the aberrations of a genuine optical system, albeit with the downsides of high computational cost and the inevitability of introducing artificial ringing artifacts into the deconvolved spectrum[1]. The spectroperfectionism

---

[1] https://hoggresearch.blogspot.com/2019/10/complexifying-optimal-extraction.html

approach has recently been demonstrated on échelle spectra for PRV purposes, achieving comparable performance as optimal extraction for the MINERVA spectrograph (Cornachione et al. 2019), a promising advance towards EPRV.

These existing techniques generally suit the vast majority of use cases, but necessarily cannot handle all science applications. For example, spatially resolved binaries with separations less than the spectrograph slit length can be acquired with alignment along a spectrograph entrance slit, nominally offering two spectra for the price of one. But data acquired in this non-standard manner would break most normal pipelines. More generally, any extended source, such as a photo-dissociation regions (PDR), Solar System planet surfaces, or resolved circumstellar Keplerian gas disks, possess an admixture of spatial and spectral information convolved together in a way that can make separation and interpretation difficult. A hypothetical general-purpose spectroscopic scene model would be enormously helpful in these contexts, but few such plug-and-play solutions exist. Spectroastrometry (Pontoppidan et al. 2011) and the recent slitless scene model for CTIO (Neveu et al. 2023) stand out as recent successes.

Brown dwarfs represent a particularly adversarial example for spectral extraction for two reasons. Their intrinsically low luminosities mean that all but the closest brown dwarfs have low signal-to-noise ratio at high spectral resolution with existing facilities. Their ultracool atmospheres contain many molecular absorption bands, in which large swaths of spectra plunge to near zero detectable flux. The combination of low signal-to-noise ratio and large chunks of missing continuum means that the mere act of identifying a spectral trace becomes difficult, breaking core assumptions in Optimal Extraction. An even more extreme example—emission line sources—lack a continuum entirely, defying the common pipeline assumption that some detectable signal resides in each pixel sequence. Heuristics for handling these special cases may exist, but they are often bespoke or require user intervention to guide the process along.

Even in EPRV, the assumptions in optimal extraction are not actually optimal, and the choices in spectroperfectionism not actually perfect. Optimal extraction generally treats the cross-dispersion direction as exactly aligned with the detector $y$ axis as illustrated in Figure 1. This computationally expedient heuristic could be expected to introduce small RV biases that grow as the spectral traces tilt and bend farther from the detector pixel alignment. As seen in the right side of Figure 1, the cross-dispersion axis (normal to the dispersion direction), tilts away from the columns. Straight-line tilted traces have the effect of polluting adjacent wavelength bins in nearly equal measure and, to first order, lowering the effective spectral resolving power. Ideally we would extract along the cross-dispersion axis ("slit-aligned"), but doing so may force us to re-grid the spectrum via intepolation, a loss mechanism that loses the native pixel uncertainties by correlating adjacent pixels.

Curved échelle traces are even more insidious. The smearing of pixels can be expected to proceed with a systematic RV shift, since the shoulder of one spectral line notices slightly more of its pixel neighbor to the left or right, depending on the extent and direction of the spectral trace's curvature. These seemingly pathological minor biases must make a difference at the 10 Si atom level, but quantifying such complex effects has been elusive in part due to the lack of an end-to-end simulation framework for échellograms.

We seek a spectrograph's *digital twin*. This term refers to a class of extreme forward modeling–a digital twin should represent all or most of the relevant behaviors of a complex system in a way that can be interrogated to build novel insights. A digital twin should resemble—as close as possible—the actual state of the system at any given moment, providing simulated data that is indistinguishable from a genuine observation. An échelle spectrograph digital twin therefore builds upon and transcends the notions of scene modeling introduced in Bolton & Schlegel (2010), by aspiring to include evermore realistic attributes of the spectrograph's imaging system into the computational machinery. A digital twin would improve upon these limitations by providing a more physically realistic generative model for the underlying spectrum.

The challenges of creating a spectrograph's digital twin are enormous. Generally only a few individual practitioners possesses the requisite knowledge of all the copious ways a spectrograph may be perturbed, and an even smaller fraction of such individuals have the capacity or desire to translate those perturbations into an evaluable forward scene model. Even if such a model could be written down and codified into a program, evaluating that program is destined to be computationally expensive owing to a litany of `for` loops on millions of pixel-based simulations. Any likelihood function would need to be reëvaluated as a Bayesian inverse problem to determine the settings of the tunable scene model. The spectrograph state, sky emission lines, and underlying target star spectrum may have tens of thousands of variables needed to adequately describe the spectrogram. Tuning tens of thousands of non-linear parameters in this complicated forward model would appear hopelessly intractable. Even if one could somehow lin-
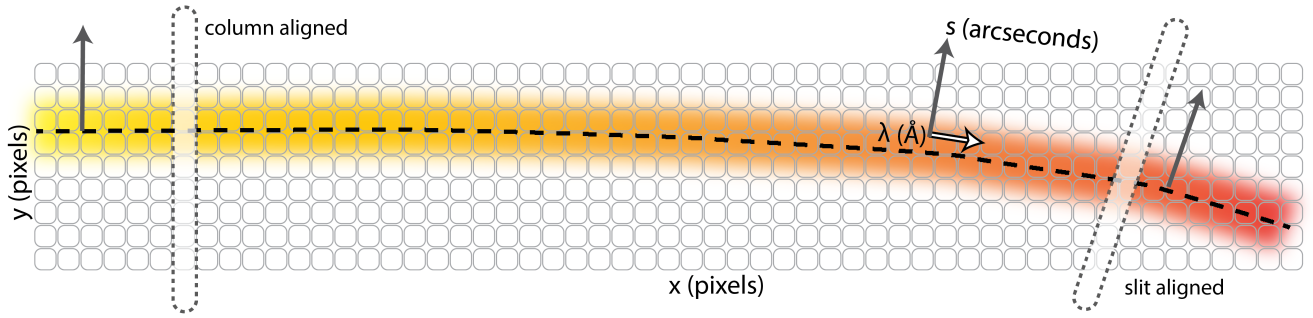
**Figure 1.** Spectral trace geometry for curved échellograms. A spectral trace that has the dispersion axis $\hat{\boldsymbol{\lambda}}$ perfectly aligned with the pixel $\hat{\boldsymbol{x}}$ row vector ("column aligned") can safely undergo "optimal extraction" (Horne 1986). But as the trace curves, the cross-dispersion axis $\hat{\boldsymbol{s}}$ tilts away from the verical columns $\hat{\boldsymbol{y}}$. An ideal extraction box may be aligned with the cross-dispersion axis ("slit aligned" for a slit-based instrument). The right choice for extraction becomes complicated in this scenario and forms the basis of this paper.

earize a large subset of these parameters via linear algebra (Bolton & Schlegel 2010), correlated residual structures would inevitably predominate the likelihood calculation, hampering the application of computationally expedient chi-squared algorithms.

Machine learning innovations make digital twins newly possible. First, and most importantly, machine learning frameworks such as PyTorch (Paszke et al. 2019), Tensorflow (Abadi et al. 2015), and JAX (Bradbury et al. 2018) offer end-to-end atomatic differentiation. This key enabling technology makes it possible to tune an unlimited number of non-linear parameters through the process of vectorized Stochastic Gradient Descent (SGD, Ruder 2016), or simply *backprop*. Second, these frameworks all offer seamless integration with hardware accelerators, including Graphical Processing Units (GPUs) and Tensor Processing Units (TPUs), that can yield 60-100× speedups with no code change. Third, neural network architectures and/or kernel methods offer a strategy for learning patterns in correlated residual structures, which has the effect of making likelihood functions tolerant to inevitable imperfections in the forward model. All together these and other modern computational innovations unlock a truly new category of ambition for our data analysis, bringing a full spectrograph forward model into reach.

In this work, we introduce a new interpretable machine learning framework for the analysis of 2D échellograms. We write down and illustrate the the design of the forward model in §2, including a breakdown into sky, target, and background steps. These steps are generally agnostic to the spectrograph design, and could apply to either a slit-based or fiber-fed instrument with some tuning. We choose to design our protoype on archival Keck NIRSPEC spectra of faint ultracool brown dwarf spectra, a category already identified as particularly vulnerable to poor optimal extraction. In §4, we first il-

lustrate the injection and recovery of known signals in synthetic spectrograms, running the synthetic spectra through facility pipelines as if they were bona fide observations. We then show the results on genuine data in §4, comparing the end-to-end performance of our technique with facility pipelines. Finally in §6, we discuss the tradeoffs of the technique and ways in which it could be extended, and design considerations for other applications. We discuss economic considerations of applying the framework, weighing the computational cost against the lifecycle cost of traditional spectral acquisition and analysis. Throughout the work we dispell common criticisms of machine learning as a black box, since our method surgically adopts the interpretable components of the multi-faceted technology while largely eschewing fashionable-albeit-uninterpretable neural network architectures. We provide the Python implementation as an open source code, `ynot`[2]. We intend the framework to serve as community destination for modular experimentation on digital twins for astronomical spectroscopy.

## 2. METHODOLOGY: MODELING 2D PIXELS

### 2.1. *Autodiff as a central design choice*

Our method has one uncompromising architectural decision that drives many subsequent design features: we must have automatic differentiation. Without *autodiff*, the optimization of our spectrogram scene model would be intractable. The need for autodiff restricts our adoption of programming language to only one of three or four popular Machine Learning (ML) frameworks, since these ML frameworks offer autodiff by construction. Which of these several ML frameworks to select is a matter of taste–we implemented our method in `PyTorch`, and anticipate a future prototype may be

---

[2] https://github.com/gully/ynot

written in `JAX`. Both of these provide partial derivatives of the likelihood function $\mathcal{L}$ with respect to all input parameters of the scene model:

$$\frac{\partial \mathcal{L}}{\partial \vec{\theta}} \qquad (1)$$

where $\vec{\theta}$ represents the vector of tunable parameters that we will soon encounter, and will occupy the lion's share of this section. We anticipate the length of $\vec{\theta}$ to number in the tens of thousands of parameters. Models that can provide Equation 1 are called *differentiable*. One consequence of requiring our models to be differentiable is that we cannot use any outside libraries: `astropy`, `Scipy`, and `numpy` would break the ability of `PyTorch` to deliver Equation 1. The constraint to never-leave-`PyTorch` makes an already difficult task even more difficult, but possible.

### 2.2. *The Shape of Data and Likelihood*

Data from a raw spectrogram in its purest form contains $d_{ij}$, where $i$ and $j$ are the fixed pixel indices in the $x$ and $y$ directions respectively, and $d$ is the digitized datum value in analog-to-digital units (ADUs). The full frame image of an échellogram is stored as an $(N \times M)$ matrix, which we will call $\mathbf{D}$ to emphasize that this object represents *data*:

$$\mathbf{D} = \begin{bmatrix} d_{11} & \dots & d_{1N} \\ \vdots & \ddots & \vdots \\ d_{M1} & \dots & d_{MN} \end{bmatrix} \qquad (2)$$

Our digital twin seeks to make an interpretable forward model $\mathbf{M}(\vec{\theta})$ that has the exact same shape as $\mathbf{D}$. We can then write down the data-model residual matrix and write the familiar $\chi^2$ likelihood function:

$$\mathbf{R}(\vec{\theta}) = \mathbf{D} - \mathbf{M}(\vec{\theta}) \qquad (3)$$

$$\mathcal{L} = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} \left( \frac{r_{ij}(\vec{\theta})}{\sigma_{ij}} \right)^2 \qquad (4)$$

where the $\sigma_{ij}$ are the nominally fixed per-pixel uncertainty estimates. Written in this way, we can see that every pixel can—at least in principle—inform every model parameter. Equation 1 senses what infinitesimal changes to each model parameter would result in a closer match to the data, a higher likelihood $\mathcal{L}$ (*i.e.* a lower $\chi^2$). This statement encodes the hope of a digital twin, that by leveraging all of the raw data, our constraints on the state of the spectrograph can be as well-informed as possible.

### 2.3. *Cartesian pixels to spectrum-centric coordinates*

The chief operation of a spectrograph is to disperse overlapping light into different spatial coordinates. Spectral extraction aims to *undo* that process, seeking a lookup table of $x$ coordinates to wavelength coordinates $\lambda$. Here we introduce the 2D analog of wavelength calibration, seeking a mapping from $(x, y)$ pixels to some as-yet-unspecified spectrograph-centric coordinate system. Spectrograms encode two unseen native spectral coordinates, the dispersion axis $\lambda$ and the cross-dispersion axis that we will designate as $s$. Diffractive optical elements such as échelle gratings have an additional special property of repeating spectral coordinates into adjacent échelle orders, indexed as $m$, according to the grating equation (James 2012). The mapping of the physics-based coordinates $(\lambda, s, m)$ to the detector $(x, y)$ Cartesian pixel coordinate system can be abstracted as some enormously complicated continuous vector-valued function $G$:

$$G(x, y) = (\lambda, s, m) \qquad (5)$$

Nature does not know about $G$, but instead its similarly complicated inverse: $G^{-1}(\lambda, s, m) = (x, y)$. The choice of how to obtain $G$ (or its inverse) forms our second key architectural design consideration (after requiring autodiff). Hypothetically, $G$ could be derived from the spectrograph's optical design software, validated with in-lab optical metrology, and adjusted for on-sky performance. Common software already takes $(\lambda, s, m)$ as inputs and returns focal plane coordinates as outputs, so this physically-motivated $G^{-1}$ appears ideal for a digital twin. But optical modeling comes with some challenges and tradeoffs. First, most optical designs are conducted in a proprietary program such as Zemax™, which is generally incompatible with Eq. 1: you cannot take partial derivatives of a Zemax model. An suitable alternative could be to rewrite your entire optical model in a differentiable optical modeling framework such as `morphine` (Pope et al. 2021) or $\partial$Lux (Desdoigts et al. *in prep*).

For now, we simply treat $G$ as a phenomenological parametric model, and emphasize that the `ynot` code is written in a future-proof way, such that the $G$ module can be hot-swapped with a drop-in-replacement physics-based model if/when one becomes available.

### 2.4. *Breaking the scene into local and global components*

In practice, not all pixels inform all parameters. So our third key architectural design choice is to separate physical and instrumental effects as being either global or local. Local physical or instrumental effects are much

easier to model than global ones, since their computation can be restricted to a subset of all pixels. We identify a science target frame as consisting of 5 distinct constituent layers, only some of which touch all pixels. For example, all pixels exhibit a distinct global bias and thermal background response: a changing detector temperature generally affects all pixels. Conversely, only some pixels "see" the sky background, and an even smaller subset sense the thin trace of the science target spectrum. Scattered light and/or internal optical reflections are somewhere in-between, depending on the scale-length and angles of scattering. Additional binary layers for dead pixels and cosmic ray masks can be treated similarly, for a total of 7 distinct physical or instrumental layers that may affect any given pixel. The model can be re-written as some separable combination of these constituent parts, and in the process we can designate the native coordinate system of each component:

$$
\begin{aligned}
\mathsf{M}(\vec{\theta}) = {} & \mathsf{M}_{\mathrm{b}}(x, y) \quad \textit{bias term} \\
& + \mathsf{M}_{\mathrm{t}}(x, y) \quad \textit{thermal response} \\
& + \mathrm{M}_{\mathrm{sky}}(\lambda, s, m) \quad \textit{sky emission} \\
& + \mathrm{M}_{\mathrm{targ}}(\lambda, s, m) \quad \textit{science target} \\
& + [\texttt{scattered light term}] \\
& + \dots \quad \textit{miscellaneous}
\end{aligned}
\tag{6}
$$

The bias and thermal response (*i.e.* dark) terms depend chiefly on the state of the spectrograph's readout electronics and detector temperature, and are usually considered fixed and determined *a-priori*. Our spectrograph's digital twin forward scene model will allow these terms to take on a parameteric form described in Appendix B, amounting to a relatively negligible effect for most applications. We ignore the scattered light term and other miscellaneous terms for now as well, a fair assumption for many spectrographs and use cases.

The sky emission and target terms are special. First, they reside in spectrograph coordinates, which can be converted to spatial pixels with $G^{-1}$. Second—as emphasized above—their reach is local. We therefore adopt *sparse matrices* to represent the sky and science target terms. We assign the slightly nonstandard convention of denoting sparse matrices with the un-bolded symbol M, and their counterpart *dense matrices* with the bold version. A sparse matrix can be pictured as resembling a vector, since it stores only the non-zero elements as a trio of coordinates-and-value, and assumes zeros for all other entries. We can visualize an example sparse matrix as an augmentation to Equation 2:

$$
\mathrm{M} =
\begin{bmatrix}
(7, 2, m_{72}) \\
\vdots \\
(i, j, m_{ij}) \\
\vdots \\
(3, 5, m_{35})
\end{bmatrix}
\tag{7}
$$

## 3. HOW TO REPRESENT THE TARGET SPECTRUM

Writing down a model for the science target spectrum represents both the chief science objective, and nearly circular reasoning. We need to know the underlying spectrum of the target star in order to fully model the scene, but if we had perfect knowledge of the target star we would not need to model the scene, nor take any data in the first place. Instead we have to infer the spectrum. The right choice depends strongly on one's science goals, and there does not exist a single best choice for all use cases. Here we narrate three pathways.

We could simply have a non-parameteric super-resolution highly flexible model for the underlying spectrum that gets iteratively updated and solved for. This strategy works adequately (Bolton & Schlegel 2010).

Relatedly, we could employ a differentiable tunable model, such as `wobble` (Bedell et al. 2019) or *blasé* (Gully-Santiago & Morley 2022).

Finally, if the end-goal of the science analysis is to fit a tunable model to the spectrum, the data-model comparison could simply take place directly in the 2D échellogram pixels, instead of ever acquiring the 1D extracted spectrum as an intermediate data product.

We choose a combination of the first two options, most resembling a crossover of Bolton & Schlegel (2010) to Bedell et al. (2019). For the rest of the paper we will simply assume the target is a stellar point source, though the technique is relatively agnostic to the astrophysics of the source properties. We write down the parameterized stellar spectrum as a contiguous string of adjacent and overlapping 2D Gaussians, each located at wavelength coordinate $\lambda_k$.

$$
f_k(\lambda_k) = A_k e^{-\left(\frac{\lambda - \lambda_k}{\epsilon}\right)^2}
\tag{8}
$$

> **How to represent the target spectrum**
> - How to represent the sky spectrum
> - How to represent the target PSF
> - How to represent the sky spatial extent
> - How to represent the slit

### 3.1. *Constructing a Resilient Likelihood*
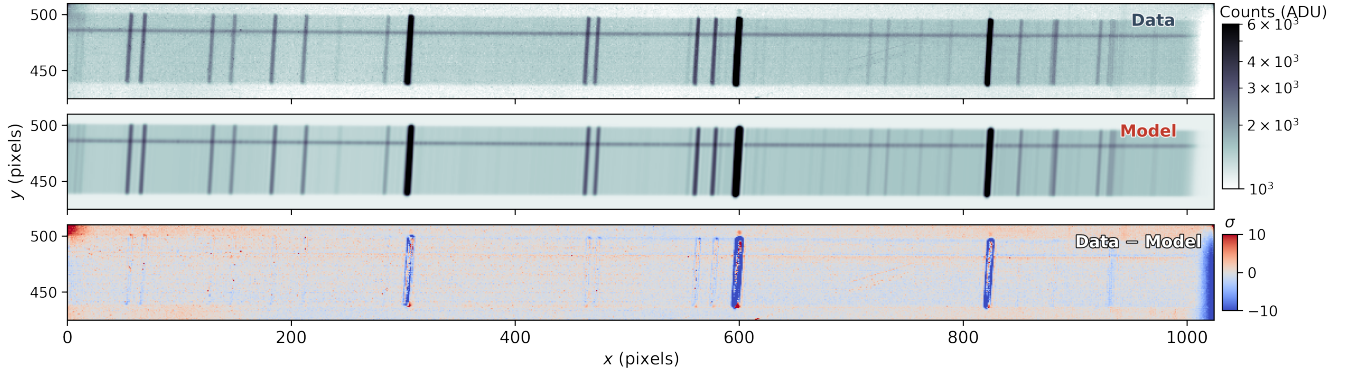
**Figure 2.** Scene model for a portion of a Keck NIRSPEC échellogram centered on a single spectral order. The sky lines, échelle order boundaries, and stellar point source spectral trace appear as familiar features. Skyline emission lines exhibit the greatest residual amplitudes. The residual also exhibits minor structures near the photon noise limit, such as detector pattern noise. Vignetting can be seen at the right edge and upper left corner.
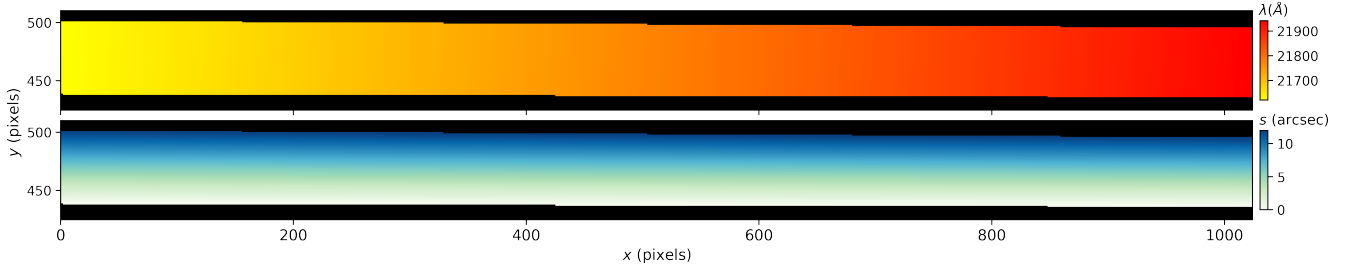


**Figure 3.** Scene Coordinates. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

**The need to joint model, regularize, address outliers**

   - The likelihood function and per-pixel uncertainties

   - Arcs: sparsely encode both wavelength and slit position

   - Flats: encode just slit position

   - Darks: encode just background

   - Target spectra: encode wavelength, slit position, target position

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

## 4. RESULTS 1: TRAINING ON SYNTHETIC DATA WITH INJECTION/RECOVERY TESTS

**Injection/recovery test with noisy data: generating fake data**

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu

nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

## Initializing of the model and optimization setup

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

## Training computational performance

- Number of epochs
- Batching/sparsity

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

## Best fit model comparison 1: injection/recovery of initial parameters

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis.

Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

## Best fit model comparison 2: spectrum as unbinned, weighted samples

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

## 5. RESULTS 2: TRAINING ON REAL DATA

## Introduction to real data

- Data-preprocessing and heuristics

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

## Outcome: reduced spectrum as unbinned, weighted samples

- SNR improvement compared to previous methods (head-to-head)

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

## 6. DISCUSSION

**Philosophy: Each previous spectrum should inform future spectra**

Treat the instrument as a breathing, dynamic system

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

**The promise for EPRV**

- Simulation of minor RV shifts
- Simulation of sub-pixel flat fields
- Tracking spectrograph state across decades of operation

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

**Ability to repurpose non-standard data (variable data quality)**

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

**Conceivable extensions**

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

**Table 1.** Selection of open source codes for 2D echelle data reduction

| Code | Instrument | Reference | Language | Note |
|------|-----------|-----------|----------|------|
| | | échelle | | |
| PLP | IGRINS | Sim et al. (2014); Lee (2015) | Python | |
| PyReduce | HARPS, UVES | Piskunov et al. (2021) | Python | |
| APERO | SPIRou | Cook et al. (2022) | Python | |
| | | Long-slit | | |
| Spectractor | CTIO | Neveu et al. (2023) | Python | Scene Modeling |

NOTE—Some codes are applicable to multiple instruments, and some instruments have multiple codes.

## APPENDIX

## A. COMPARISON TO OTHER FRAMEWORKS

Hundreds or possibly thousands of 2D extraction pipelines have been written and rewritten to suit the disparate needs of each unique spectrograph. Table 1 of Cook et al. (2022) lists 18 échelle spectrographs and their data reduction pipelines. Many but not all of those pipelines are open source. Here we list a subset of recent open source pipelines for échelle spectra reduction.

Few approaches have attempted to combine spectral scene modeling in combination with machine learning techniques. Xu et al. (2020) employed a 2D PSF convolution with neural networks on a demonstration with multi-object spectra from LAMOST. While promising, the code is not open source and the applications limited to inferring the field distortion, but not extending to subsequent extraction steps.

Cornachione et al. (2019) developed spectroperfectionism for the MINERVA spectrograph, finding similar performance as optimal extraction.

## B. FLAT FIELDS, DARK, AND ARC FRAMES

Every spectrograph is different, but several common categories of raw data frames can be expected: flat field frames $\mathbf{D}_{\mathrm{flat}}$, dark frames $\mathbf{D}_{\mathrm{dark}}$, wavelength calibration frames $\mathbf{D}_{\mathrm{arc}}$, and science target frames $\mathbf{D}_{\mathrm{targ}}$. For our purposes, observations of standard-star frames reside in the science target frame category, since they share the same observational configuration. Most of the main text of this document focused on the modeling of the science target frame, and so we simply dropped any subscripts and referred to this auspicious data as simply $\mathbf{D}$. All of these calibration frames inform the state of the spectrograph, and can and should be used as such. The `ynot` framework provides specialized modeling for these datatypes.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., et al. 2015, Google White Paper. https://www.tensorflow.org/

Ahmed, S., Hall, D., Crowley, C., et al. 2020, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 11454, X-Ray, Optical, and Infrared Detectors for Astronomy IX, ed. A. D. Holland & J. Beletic, 114540S, doi: 10.1117/12.2562162

Bedell, M., Hogg, D. W., Foreman-Mackey, D., Montet, B. T., & Luger, R. 2019, AJ, 158, 164, doi: 10.3847/1538-3881/ab40a7

Bolton, A. S., & Schlegel, D. J. 2010, PASP, 122, 248, doi: 10.1086/651008

Bradbury, J., Frostig, R., Hawkins, P., et al. 2018, JAX: composable transformations of Python+NumPy programs, 0.2.5, ". http://github.com/google/jax

Cook, N. J., Artigau, É., Doyon, R., et al. 2022, PASP, 134, 114509, doi: 10.1088/1538-3873/ac9e74

Cornachione, M. A., Bolton, A. S., Eastman, J. D., et al. 2019, PASP, 131, 124503, doi: 10.1088/1538-3873/ab4103

Gully-Santiago, M., & Morley, C. V. 2022, ApJ, 941, 200, doi: 10.3847/1538-4357/aca0a2

Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, Nature, 585, 357, doi: 10.1038/s41586-020-2649-2

Horne, K. 1986, PASP, 98, 609, doi: 10.1086/131801

Hunter, J. D. 2007, Computing in Science and Engineering, 9, 90, doi: 10.1109/MCSE.2007.55

James, J. 2012, Spectrograph Design Fundamentals

Jones, E., Oliphant, T., Peterson, P., et al. 2001–, SciPy: Open source scientific tools for Python. http://www.scipy.org/

Lee, J.-J. 2015, plp: Version 2.0, doi: 10.5281/zenodo.18579

Mahadevan, S., Ramsey, L. W., Terrien, R., et al. 2014, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 9147, Ground-based and Airborne Instrumentation for Astronomy V, ed. S. K. Ramsay, I. S. McLean, & H. Takami, 91471G, doi: 10.1117/12.2056417

McKinney, W. 2010, in Proceedings of the 9th Python in Science Conference, ed. S. van der Walt & J. Millman, 51 – 56

Neveu, J., Brémaud, V., Antilogus, P., et al. 2023, arXiv e-prints, arXiv:2307.04898, doi: 10.48550/arXiv.2307.04898

pandas development team, T. 2020, pandas-dev/pandas: Pandas, latest, Zenodo, doi: 10.5281/zenodo.3509134

Paszke, A., Gross, S., Massa, F., et al. 2019, arXiv e-prints, arXiv:1912.01703. https://arxiv.org/abs/1912.01703

Paszke, A., Gross, S., Massa, F., et al. 2019, in Advances in Neural Information Processing Systems 32, ed. H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Curran Associates, Inc.), 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Piskunov, N., Wehrhahn, A., & Marquart, T. 2020, arXiv e-prints, arXiv:2008.05827. https://arxiv.org/abs/2008.05827

—. 2021, A&A, 646, A32, doi: 10.1051/0004-6361/202038293

Piskunov, N. E., & Valenti, J. A. 2002, A&A, 385, 1095, doi: 10.1051/0004-6361:20020175

Pontoppidan, K. M., Blake, G. A., & Smette, A. 2011, ApJ, 733, 84, doi: 10.1088/0004-637X/733/2/84

Pope, B. J. S., Pueyo, L., Xin, Y., & Tuthill, P. G. 2021, ApJ, 907, 40, doi: 10.3847/1538-4357/abcb00

Pérez, F., & Granger, B. E. 2007, Computing in Science and Engineering, 9, 21, doi: http://dx.doi.org/10.1109/MCSE.2007.53

Ruder, S. 2016, arXiv e-prints, arXiv:1609.04747. https://arxiv.org/abs/1609.04747

Sim, C. K., Le, H. A. N., Pak, S., et al. 2014, Advances in Space Research, 53, 1647, doi: 10.1016/j.asr.2014.02.024

Vorobiev, D., Ninkov, Z., Caldwell, D., & Mochnacki, S. 2018, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 10698, Space Telescopes and Instrumentation 2018: Optical, Infrared, and Millimeter Wave, ed. M. Lystrup, H. A. MacEwen, G. G. Fazio, N. Batalha, N. Siegler, & E. C. Tong, 106985J, doi: 10.1117/12.2313727

Xu, J., Yin, Q., Guo, P., & Zheng, X. 2020, MNRAS, 499, 1972, doi: 10.1093/mnras/staa2883

Zechmeister, M., Anglada-Escudé, G., & Reiners, A. 2014, A&A, 561, A59, doi: 10.1051/0004-6361/201322746