

# A public data set of Ariel simulated observations for developing exoplanetary atmosphere data reduction pipelines

Lorenzo V. Mugnai<sup>1</sup>,<sup>2,3,4</sup>★ Kai Hou Yip,<sup>5</sup>★ Andrea Bocchieri<sup>1</sup>,<sup>3</sup> Andreas Papageorgiou,<sup>1</sup> Virginie Batista,<sup>6</sup> Orphée Faucoz,<sup>7</sup> Angèle Syty,<sup>6</sup> Tara Tahseen<sup>1</sup>,<sup>2</sup> Enzo Pascale<sup>3</sup> and Ingo Waldmann<sup>2</sup>

<sup>1</sup>School of Physics and Astronomy, Cardiff University, Queens Buildings, The Parade, Cardiff CF24 3AA, UK

<sup>2</sup>Department of Physics and Astronomy, University College London, Gower Street, London WC1E 6BT, UK

<sup>3</sup>Dipartimento di Fisica, La Sapienza Università di Roma, Piazzale Aldo Moro 5, I-00185 Roma, Italy

<sup>4</sup>INAF – Osservatorio Astronomico di Palermo, Piazza del Parlamento 1, I-90134 Palermo, Italy

<sup>5</sup>Department of Physics, King's College London, University of London, Strand, London WC2R 2LS, UK

<sup>6</sup>Sorbonne Université, CNRS, UMR 7095, Institut d'Astrophysique de Paris, 75014, Paris, France

<sup>7</sup>CNES, Centre national d'études spatiales, 75001, Paris, France

Accepted 2026 April 28. Received 2026 March 31; in original form 2025 September 18

## ABSTRACT

Detecting and characterizing exoplanet atmospheres remains challenging because atmospheric signals can be comparable to residual noise and instrumental/astrophysical systematics. Spectral features span from a few ppm for small planets up to  $\sim 10^3$  ppm for warm/hot giants, while high-quality *James Webb Space Telescope* (JWST) time-series spectroscopy typically reaches  $\sim 10$ –50 ppm (occasionally  $\sim 100$ –200 ppm in the presence of stellar variability or stronger systematics), making correlated noise across temporal and spectral dimensions a key limitation. With JWST delivering an increasing volume of high-precision transmission spectra, and Ariel set to extend this to a homogeneous survey of  $\sim 10^3$  exoplanet atmospheres, robust benchmarking resources with known ground truth are essential to develop and validate data-driven (including ML-based) detrending approaches. As a major step towards this goal, we use ExoSim2 and TauREx to generate one of the most comprehensive public data sets based on the current payload design of the ESA Ariel mission, specifically intended to benchmark detrending algorithms. We also provide a deep neural network baseline for time-series reduction, and use it to highlight the limitations of ML based detrending methods, i.e. the risks posed by data set shift when observed distributions diverge from those of the training set, a scenario likely to arise in real observations. This data set is featured in the Ariel Data Challenge 2024 on Kaggle and has been field-tested for robustness and simulation fidelity. By making these resources publicly available, we aim to support the community in developing, comparing, and stress-testing scalable and reliable methods for exoplanet transmission spectroscopy.

**Key words:** exoplanet atmosphere – machine learning – data detrending – denoising.

## 1 INTRODUCTION

Recent and ongoing space missions represent a major advancement in the atmospheric characterization of exoplanets. The *James Webb Space Telescope* (JWST) mission (T. P. Greene et al. 2016) is now delivering high-precision spectroscopic observations of transiting and directly imaged exoplanets, with typical noise levels of tens of ppm, enabling detailed constraints on atmospheric composition, temperature structure and cloud properties (e.g. M. Radica et al. 2023; A. L. Carter et al. 2024; R. C. Challener et al. 2025; N. Espinoza et al. 2025; J. Taylor et al. 2025). These observations are opening new frontiers in the study of planetary atmospheres across a wide range of planetary regimes. The Ariel mission (Atmospheric Remote-sensing Infrared Exoplanet Large-survey) of ESA, scheduled for launch in 2029 (G. Tinetti et al. 2018), is designed as a dedicated atmospheric survey. Its primary objective is to observe a large and statistically significant sample of exoplanets, spanning super-Earths to gas giants, in order to enable comparative planetology. Ariel is expected to characterize of order  $\sim 1000$  planetary atmospheres with a homogeneous observing strategy, providing a unique data set to study population-level trends in atmospheric chemistry and physics.

\* E-mail: [mugnail@cardiff.ac.uk](mailto:mugnail@cardiff.ac.uk) (LVM), [kai.hou.yip@ucl.ac.uk](mailto:kai.hou.yip@ucl.ac.uk) (KHY)

The scale of exoplanet atmospheric characterization is entering a new era. While *JWST* provides detailed spectra of individual targets, Ariel will survey approximately 1000 exoplanet atmospheres. This transition from targeted observations to large-scale surveys requires scalable, automated analysis approaches capable of maintaining uniform quality standards across diverse planetary systems.

Public benchmark data sets with known ground truth are essential preparation for this data-intensive regime. Unlike real observations where atmospheric properties remain unknown, simulated data sets enable rigorous quantification of systematic biases, assessment of noise propagation, and direct comparison of methodological approaches under controlled conditions. Such resources accelerate technique development through transparent community-wide benchmarking.

The Ariel Data Challenge series (K. H. Yip et al. 2022; N. Nikolaou et al. 2023; Q. Changeat & K. H. Yip 2023) exemplifies this community-driven approach, connecting planetary scientists, instrument experts, and data scientists to tackle key challenges in exoplanet characterization. The 2024 edition (K. H. Yip et al. 2024) addresses the need for realistic training data by introducing a large-scale public data set of simulated raw observations from the Ariel mission, specifically designed to benchmark detrending algorithms against realistic instrumental systematics and noise sources. This data set was generated by coupling ExoSim2 (L. V. Mugnai et al. 2025) and TauREx3 (A. F. Al-Refaie et al. 2021), incorporating the latest Ariel mission specifications, noise maps derived from publicly available *JWST* detector calibrations, and diverse atmospheric compositions. The scale and complexity of this resource provide an invaluable platform for developing and testing innovative data reduction techniques essential for future observational programmes.

Developing and optimizing bespoke data-reduction pipelines for exoplanet characterization is essential in the lead-up to forthcoming missions. Such a pipeline comprises a suite of analysis techniques, implemented as specialized algorithms, that transform raw photometric and spectroscopic time-series data into scientifically usable products, most notably the planetary spectrum as a function of wavelength. An ideal pipeline should behave as an unbiased estimator that minimizes post-processing noise by detrending systematic effects without introducing bias. This requirement is particularly stringent for the Ariel mission, given its ambitious objectives in comparative planetology and its commitment to a uniform observational strategy, which demands precise and accurate results for a self-consistent, homogeneous characterization across its exoplanet sample. Currently, several data reduction pipelines with distinct methodologies exist, and the *JWST* pipelines serve as a notable example of this diversity, with multiple independently developed pipelines introduced in the ERS context (e.g. L. Alderson et al. 2023; Z. Rustamkulov et al. 2023; E.-M. Ahrer et al. 2023; A. D. Feinstein et al. 2023) as well as additional community pipelines released in the following years (e.g. M. Holmberg & N. Madhusudhan 2023; S. Sarkar et al. 2024).

The coexistence of multiple *JWST* pipelines has also enabled direct pipeline-to-pipeline comparisons on the same data sets. In several ERS studies, independent reductions and analysis frameworks have been applied to the same observations, generally finding consistent spectra within uncertainties while highlighting that specific methodological choices (e.g. systematics modelling, outlier rejection, background treatment, time-dependent instrument trends) can still induce measurable differences (e.g. L. Alderson et al. 2023; Z. Rustamkulov et al. 2023; E.-M. Ahrer et al. 2023; A. D. Feinstein et al. 2023; S. P. Schmidt et al. 2025; L. Welbanks et al. 2025).

Machine learning (ML) and deep learning (DL) approaches have been successfully applied in various imagery data domains for denoising (M. Elad, B. Kavar & G. Vaksman 2023), reconstruction (J. Su, B. Xu & H. Yin 2022), and even generating new images (C. Zhang et al. 2023; F. Bie et al. 2025; S. R. Dubey & S. K. Singh 2024). Recent studies have explored the use of ML techniques to denoise astronomical data in different forms, such as time-series light-curve data (J. G. Ingalls et al. 2016; J. E. Krick et al. 2020; M. Morvan et al. 2020, 2022; N. Nikolaou et al. 2023) and imagery data (A. Vojtekova et al. 2021; S. F. Sweere et al. 2022; H. Park et al. 2024).

However, most of these studies have focused on either time-series (light curves) or images from photometry, while actual transit observations are 3D cubes that are acquired as a time-series of still images across a wide wavelength range. This combination results in an exceptionally rich information content within a single observation. The aforementioned approaches have not utilized the full extent of this information, inevitably neglecting crucial spatial and spectral information about the noise characteristics. Past studies have indicated that incorporating temporal information (light curves) from simulated *HST* (K. H. Yip et al. 2020) and *JWST* data (Q. Changeat et al. 2024) can mitigate some of the temporal aspects of noise effects (such as star spots), and could yield better constraints on downstream tasks such as retrieving the atmospheric parameters.

As a first attempt at building a neural network-based data reduction and extraction pipeline, we trained a CNN-based neural network as a benchmark baseline. Since machine learning models are prone to overfitting to training data, it is important that our benchmark data account for this tendency. To thoroughly test any pipeline, whether parametric, machine learning, or hybrid, we have designed a data partition that explicitly includes out-of-distribution test examples. This approach exposes challenges that models will likely face when applied to real data, such as interpreting spectra from unknown exoplanets with unknown composition and atmospheric properties, known as ‘data shift’ or ‘domain shift’ in the Machine Learning and Artificial Intelligence community. This allows us to provide an estimate of the model’s performance on both training data and data that are unknown to us.

The structure of this paper is organized as follows: Section 2 outlines the various components involved in data generation and the validation processes that occur before commissioning. Section 3 details the data partitioning methodology, which divides the data set into traditional training and testing subsets. In Section 4, we present the implementation of the CNN<sup>1</sup>-based data reduction pipeline and our proposed metric to evaluate its performance, followed by an analysis of the results and performance metrics on the data sets

<sup>1</sup>Convolutional Neural Network, more explanation in the Section 5.3.

in Section 5. In Section 6, we discuss our methodology limitations and our results. Finally, Section 7 concludes with a discussion of our findings, future research directions, and the challenges that lie ahead.

The training data set produced for this paper is publicly available on the Kaggle platform.<sup>2</sup>

## 2 DATA DESCRIPTION

The simulated data are generated using the publicly available end-to-end instrument simulator ExoSim2<sup>3</sup> (L. V. Mugnai et al. 2025). ExoSim2 models the full Ariel observing chain, including the telescope throughput, the dispersive optics, the detector response, and the noise sources, and outputs realistic time-series detector images for a given target and observing scenario. To build physically consistent simulations, ExoSim2 is coupled with (i) the atmospheric retrieval and forward-modelling framework TauREx 3 (A. F. Al-Refaie et al. 2021), which provides the wavelength-dependent planetary transmission spectrum used as the astrophysical input signal, and (ii) the physical optical simulator PAOS (A. Bocchieri, L. V. Mugnai & E. Pascale 2024), which generates realistic, wavelength-dependent Ariel point-spread functions (PSFs) and optical aberrations. Using the three tools together ensures that both the astrophysical signal (planet spectrum) and the instrument response (optics/PSF and detector-level systematics and noise) are treated consistently, enabling the generation of mock images representative of Ariel observations. For a complete description of the simulation process and Ariel's specificities, please refer to the ExoSim2 documentation<sup>4</sup> and published literature (A. Bocchieri et al. 2025; L. V. Mugnai et al. 2025).

The following section outlines the strategies and assumptions employed in constructing the data set. In Section 2.1, we begin by discussing the model of the Ariel payload utilized for the simulated observations. In Section 2.2, we describe both the calibration products used to realistically simulate the detector response and a comprehensive list of all simulated noise sources included in the process. Once the telescope setup is described, we proceed to discuss the observed target star in Section 2.3 and the planetary models included in the simulation (Section 2.4). Finally, in Sections 2.5 and 2.6, we describe the data generation process and the validation of the final product, including statistics on the data set.

### 2.1 Payload model

For this data set, we refer to the payload configuration presented in L. V. Mugnai et al. (2025), and instead of all six channels, we use one photometer (FGS1) collecting light between 0.6 and 0.8  $\mu\text{m}$  and one spectrometer (AIRS-CH0) sensitive from 1.95 to 3.90  $\mu\text{m}$  with spectral resolving power  $R = 100$  (L. V. Mugnai et al. 2020). The effects of Ariel's design can be summarized as a combination of Photon Conversion Efficiency (PCE) and Point Spread Functions (PSFs) as described below. Additionally, we reduced the area (in pixels) of Ariel's channels to optimize storage space in the simulations. For the FGS, we used a  $32 \times 32$  pixel array instead of the original  $64 \times 64$  window, and for the AIRS, we used a  $32 \times 356$  pixel array instead of the original  $64 \times 356$  region of interest.

Because some parameters are not yet available for the flight model, using the *JWST* equivalent (see Section 2.2.1) is expected to be a realistic alternative, as Ariel relies on near-IR detector technology comparable to that adopted by *JWST*/NIRSpec, making CRDS-based dark, read-noise, flat-field, and bad-pixel maps a physically motivated proxy for detector-level behaviour. Nevertheless, differences in operating conditions and in-flight environment (e.g. temperature, readout electronics, and radiation-driven ageing) could shift the absolute noise properties and the level of correlated systematics. At the same time, we use critical elements from Ariel, such as the photon conversion efficiency and the point spread functions.

The photon conversion efficiency is the product of the telescope transmission efficiency and the detector quantum efficiency, and it describes the ability of the payload to convert light into electrons. For a detailed description of the conversion process, we refer the reader to the following: ExoSim2 (L. V. Mugnai et al. 2025), the Ariel radiometric model ArielRad (L. V. Mugnai et al. 2020), and the ExoSim2 documentation.<sup>4</sup> The PCE used is reported in Fig. 1.

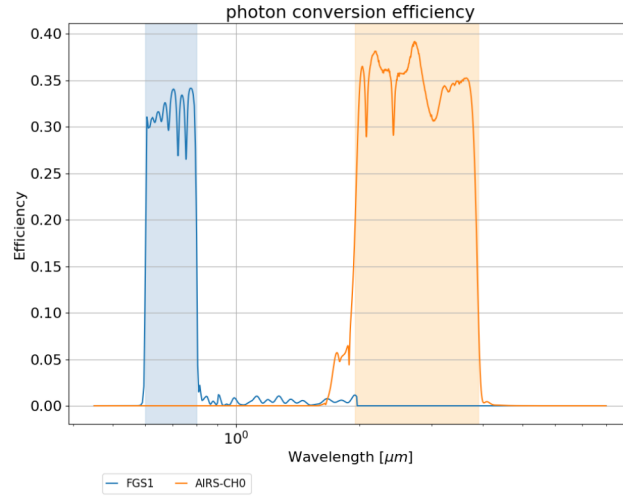
For a realistic representation of the telescope's performance, we use the PSFs already presented in other Ariel-related works (A. Bocchieri et al. 2025; L. V. Mugnai et al. 2025; V. Thuraiirethinam et al. 2025). The FGS1 PSFs are produced using the PAOS simulator (A. Bocchieri et al. 2024) for a range of wavelengths sampled by the detector, as described in L. V. Mugnai et al. (2025). For the AIRS-CH0 PSFs, we use the default ExoSim2 Airy PSF model for every wavelength sampled by a pixel. The utilized PSFs are shown in Fig. 2.

To create the final image, these PSFs are convolved with the intrapixel response function, which models the pixel response to light, as typically, a pixel is more sensitive near the centre and less toward the edges. This model is created by a default ExoSim Task using the prescription from N. Barron et al. (2007).

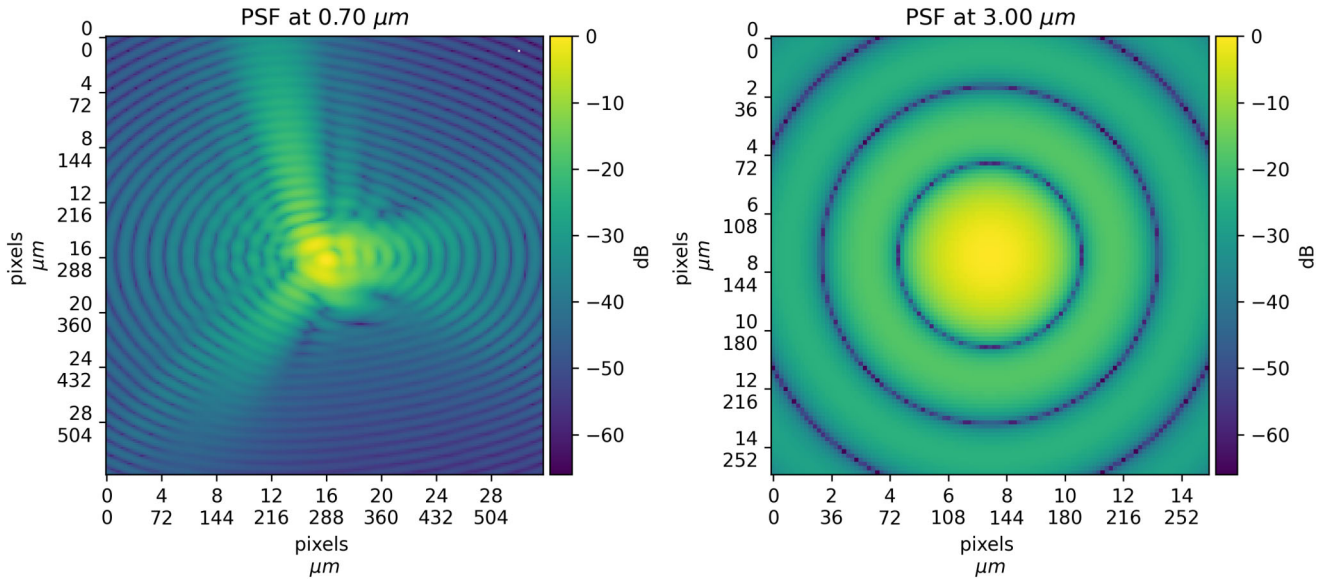
<sup>2</sup><https://www.kaggle.com/competitions/ariel-data-challenge-2024/data>

<sup>3</sup>in this work, we used code version v2.0.0-rc2, available at <https://github.com/arielmission-space/ExoSim2-public/releases/tag/v2.0.0-rc2>.

<sup>4</sup><https://exosim2-public.readthedocs.io/>



**Figure 1.** The figure shows the photon conversion efficiency (PCE) considered in this work for the two Ariel channels simulated: FGS1 and AIRS-CH0. The PCE is computed as the product of the telescope optical transmission efficiency and the detector quantum efficiency. The background coloured bands highlight the nominal wavelength ranges for each channel.



**Figure 2.** The figure reports two of the PSFs used to simulate FGS1 (left) and AIRS-CH0 (right). The PSFs are sampled at the sub-pixel level and reported in decibels normalized to the maximum value in the field. The axes report the grid into pixel and micron units. The reader may note that the FGS1 PSF is highly aberrated and complex, so it is sampled on a grid of the same size as the detector ( $32 \times 32$ ), while the AIRS-CH0 PSF is an Airy function and here is reported on a smaller grid. For details, please refer to the software documentation.

## 2.2 Detector and noise models

### 2.2.1 JWST calibration products

To simulate realistic detector maps, we utilized calibration products for NIRSpec from the *JWST* Calibration Reference Data System (CRDS). Specifically, we collected dark frames,<sup>5</sup> read noise,<sup>6</sup> flat fields,<sup>7</sup> and bad pixel maps.<sup>8</sup> These calibration files were carefully selected to ensure consistency. All CRDS reference files were taken from the NIRSpec detector NRS2. The selection of NRS2 was not driven by a specific scientific preference, but was adopted for practical consistency in this data set release.

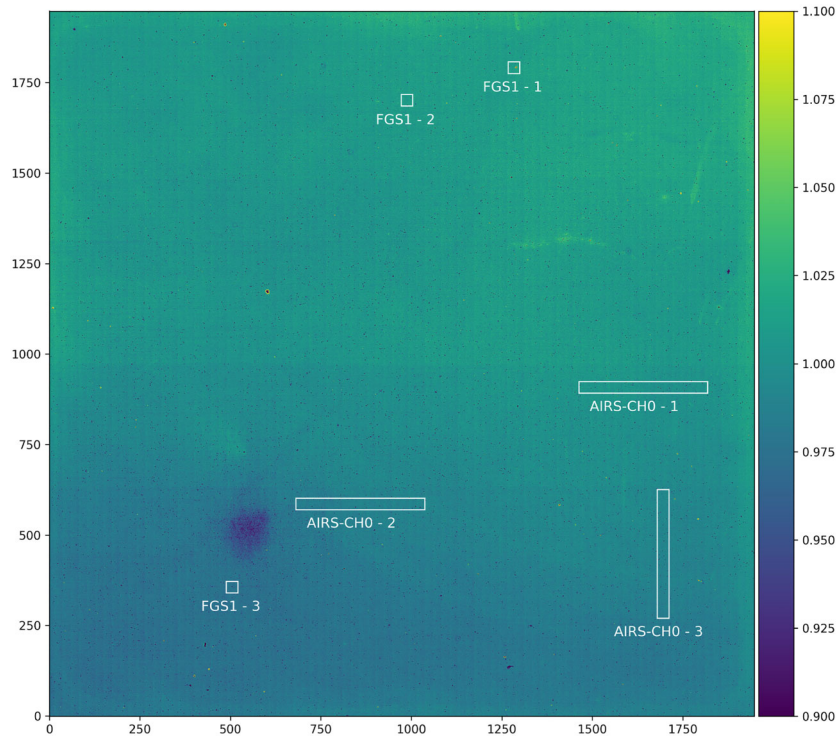
<sup>5</sup>[https://jwst-crds.stsci.edu/browse/jwst\\_nirspec\\_dark\\_0352.fits](https://jwst-crds.stsci.edu/browse/jwst_nirspec_dark_0352.fits)

<sup>6</sup>[https://jwst-crds.stsci.edu/browse/jwst\\_nirspec\\_readnoise\\_0039.fits](https://jwst-crds.stsci.edu/browse/jwst_nirspec_readnoise_0039.fits)

<sup>7</sup>[https://jwst-crds.stsci.edu/browse/jwst\\_nirspec\\_dflat\\_0002.fits](https://jwst-crds.stsci.edu/browse/jwst_nirspec_dflat_0002.fits)

<sup>8</sup>[https://jwst-crds.stsci.edu/browse/jwst\\_nirspec\\_mask\\_0049.fits](https://jwst-crds.stsci.edu/browse/jwst_nirspec_mask_0049.fits)





**Figure 3.** *JWST* flat-field variation map used for the simulation. The highlighted regions indicate the three areas selected for the each detector (FGS and AIRS-CH0) flat-field maps. This flat-field is presented as a variation from the median value. The colour scale is set between 0.9 and 1.1 to enhance detector features, masking the outlier values of bad pixels.

From this data, we extracted segments of the detector to align with the dimensions of Ariel’s FGS1 and AIRS-CH0 detectors. Several regions were selected to represent various potential detectors for Ariel. To thoroughly assess the detrending performance of any proposed pipeline, we selected one detector free of bad pixels in the illuminated area and another that includes some bad pixels or defects in the FGS1 and CH0 areas, thereby simulating a worst-case scenario.

The dark current contribution is estimated for each pixel based on the expected counts for that pixel in the input map, multiplied by the exposure time of each sub-exposure. This estimate is then subject to randomization due to Poisson noise. The read noise, on the other hand, is provided as a variance map, so for each NDR, we add a random number of counts normally distributed according to the pixel variance.

The flat fields are maps of the interpixel quantum efficiency variation, here used normalized to the median detector quantum efficiency used to compute the PCE. Finally, bad pixels are non-functional pixels that are no longer able to collect light.

To summarize how these effects combine to form the measured signal ( $S_{\text{mes}}$ ) starting from the original signal ( $S$ ), we consider the following equation:

$$S_{\text{mes}} = (S \cdot \Delta t \cdot \text{Flat}) + (\text{Dark} \cdot \Delta t) + \text{Read}, \quad (1)$$

where  $\Delta t$  is the integration time for the considered NDR, Flat is the flat-field, and Dark is the dark current map. In this equation,  $S \cdot \Delta t$  represents the signal integrated over the exposure time, and the flat-field modulates it.  $\text{Dark} \cdot \Delta t$  accounts for the dark current contribution over the same integration time. Read is the read noise, which is a random realization based on the read noise variance map. Both  $S \cdot \Delta t$  and  $\text{Dark} \cdot \Delta t$  are subject to randomization due to Poisson noise, reflecting the statistical nature of photon counting and dark current generation.

Fig. 3 shows the selected regions on the *JWST* flat-field calibration product. These regions were manually selected by visually comparing different calibration products to include a comprehensive mixture of possible detector effects. Note that this flat-field is calculated as the variation from the median value, i.e. using the flat-field divided by the median.

### 2.2.2 Pixel non-linearity and saturation

To simulate pixel non-linearity and saturation effects for the *JWST*-NIRSpec,<sup>9</sup> we utilized the published correction coefficients. We first inverted the relationship using these coefficients, applying a fifth-order polynomial to convert ideal counts to observed counts. This

<sup>9</sup>[https://jwst-crds.stsci.edu/browse/jwst\\_nirspec\\_linearity\\_0025.fits](https://jwst-crds.stsci.edu/browse/jwst_nirspec_linearity_0025.fits)

relationship was then adjusted so that the mean saturation point—defined as a 5 per cent deviation from the ideal measurement—reaches fixed levels of 100 000 counts for FGS1 and 85 000 counts for AIRS-CH0. This results in six coefficients for every pixel that map observed to corrected counts:

$$F_c = c_0 + c_1F + c_2F^2 + c_3F^3 + c_4F^4 + c_5F^5, \quad (2)$$

where  $F$  is the observed counts,  $c_n$  are the coefficients, and  $F_c$  is the corrected counts.

At these thresholds, pixel counts are capped to prevent any further increase, effectively simulating the saturation effects. Additionally, we introduced a randomization of 0.5 per cent in the recomputed fifth-order polynomial correction coefficients to mimic measured values.

### 2.2.3 Pointing jitter

ExoSim2 models the impact of pointing jitter, which refers to the inevitable vibrations and drifts of the telescope in space while observing a source, by simulating the focal plane movements during the ramp integration within a ‘sub-exposure’. The timelines utilized in this simulation are provided by Airbus Defence and Space (ADS), the prime contractor for the *Ariel* service module. Specifically, a 10-h long timeline sampled at 1 kHz is employed. A more detailed analysis of the timeline used is reported in A. Bocchieri et al. (2025) along with a detrending algorithm developed by the Ariel Consortium. Given that only one time-series is available at the time of generation, we enhance the variability of disturbances induced by this single jitter realization by randomly phase-shifting the time-series for each simulation.

The pointing jitter represents a critical systematic effect in our simulations, as it directly impacts the photometric stability of the measured signal. During the integration time, the point spread function is not stationary on the detector but continuously shifts due to the telescope line-of-sight variations. As a consequence, the incoming flux is sampled by different regions of the detector, each characterized by non-uniform intrapixel and interpixel response functions. This induces time-correlated fluctuations in the measured signal, even in the absence of intrinsic astrophysical variability. Such effects are particularly challenging to correct, as they couple spatial detector inhomogeneities with temporal variations in the pointing, producing a source of correlated noise that can be comparable in amplitude to the atmospheric features. This makes pointing jitter one of the primary challenges to be addressed in this data set, and a key driver for the development and benchmarking of robust detrending techniques.

### 2.2.4 Gain drift

The simulation of gain drift in ExoSim2 models the variation in detector gain over time and across different wavelengths. The gain noise enters the payload stability budget and needs to be at a low level for high-accuracy photometry and spectroscopy. In our simulations, the amplitude of the gain drift is randomly defined between 1 per cent and 5 per cent of the incoming signal. This range is consistent with worst-case scenarios observed in *JWST* data.

The gain drift is applied as a fifth-order polynomial function to model the drift over time and wavelength. The polynomial coefficients are randomly drawn from a uniform distribution between  $-1$  and  $1$  in both temporal and spectral directions, ensuring a comprehensive representation of potential drift behaviours.

The simulated gain drift is applied to the sub-exposures by creating a multiplicative noise map that affects each pixel’s signal. The drift is modelled separately for the spectrometer (AIRS-CH0) and the photometer (FGS1). For the spectrometer, the drift is calculated as a fifth-order polynomial function of both time and wavelength, resulting in different drift values for each column (representing different wavelengths) in the detector. This polynomial trend is determined by randomly generated coefficients within specified ranges, ensuring a realistic variability in the gain drift. The absence of correlation between adjacent pixel columns, expected to be strong in the real focal plane, is assumed as a worst-case scenario to provide a more conservative test case. The resulting polynomial values are then multiplied by the original signal to simulate the drift effect. For the photometer, the drift is modelled as a polynomial function of time only, creating a uniform drift effect across all pixels. This approach ensures that the gain drift is accurately represented for both the spectrometer and photometer channels, taking into account their specific characteristics and enhancing the robustness of the photometric measurements.

### 2.2.5 NDRs read out mode

For both channels, we implemented the Correlated Double Sampling (CDS) readout mode to sample the detector light collection ramp. This method involves recording one Non-Destructive Read (NDR) at the beginning of the ramp and another at the end. This strategy allows us to reconstruct the incoming flux using the equation:

$$\text{Flux} = \frac{\text{NDR}_1 - \text{NDR}_0}{\Delta t}, \quad (3)$$

where  $\text{NDR}_0$  is the initial NDR,  $\text{NDR}_1$  is the final NDR, and  $\Delta t$  is the integration time between them. While more advanced ramp sampling strategies (e.g. multi-accumulation schemes) are expected to be available for *Ariel*, a detailed treatment of detector readout modes is beyond the scope of this work; CDS is adopted here to provide a simple and uniform framework for data set generation and

method benchmarking, while it provides at the same time the highest SNR for Ariel's photon noise limited observations, when compared to alternative detector sampling schemes.

Each ramp cycle in both channels operates with a detector clock unit set to 10 Hz, meaning each clock cycle lasts 0.1 s. The sequence of a complete ramp is as follows:

- (i) Ground State: the ramp starts with the detector in a virtually empty state for 1 clock unit (0.1 s).
- (ii) First NDR: the first NDR ( $\text{NDR}_0$ ) is collected, which takes another clock unit (0.1 s).
- (iii) Second NDR: the detector collects photons for the integration time. For FGS1, this lasts for 1 clock unit (0.1 s) to avoid saturation. For AIRS-CH0, this lasts for 45 clock units (4.5 s) to accommodate the selected stars. The second NDR ( $\text{NDR}_1$ ) is then collected and, because the clock unit is 10 Hz, the collection lasts an additional 0.1 s.
- (iv) Ramp Reset: the ramp is reset for 1 clock unit (0.1 s).

Each ramp is thus sampled by two NDRs, and the ramps are separated by an additional 2 clock units (0.2 s) for resetting and preparation for the next ramp. Therefore, the effective integration time is 0.2 and 4.6 s for FGS1 and CH0, respectively. Each ramp lasts 0.5 and 4.9 s, respectively. So, the duty cycle is  $0.2/0.5 \sim 40$  per cent and  $4.6/4.9 \sim 93$  per cent, respectively.

### 2.2.6 Conversion to ADU

The Analogue-to-Digital Converter (ADConv) simulates the conversion of analogue signals into digital signals, represented as Analogue-to-Digital Units (ADUs). This conversion process is crucial for accurately modelling the digital readout of detectors in astrophysical instruments.

The ADC conversion operates with a 16-bit resolution, providing a high degree of precision in the digital representation of the analogue signal. The gain ( $\text{ADConv}_{\text{gain}}$ ) and offset ( $\text{ADConv}_{\text{offset}}$ ) of the ADC are set to 'auto', allowing the system to dynamically adjust these parameters based on the characteristics of the measured signal ( $S_{\text{meas}}$ ), to make use of the full values range offered by 16-bit unsigned integers. The resulting 16-bit digital signal ( $S_{16\text{bit}}$ ) is calculated as:

$$S_{16\text{bit}} = \lfloor \text{ADConv}_{\text{gain}} (S_{\text{meas}} - \text{ADConv}_{\text{offset}}) \rfloor, \quad (4)$$

where the floor function  $\lfloor \cdot \rfloor$  represents the integer part of the scaled signal. This approach ensures that the conversion process can accommodate a wide range of signal intensities, capturing the variability inherent in astrophysical observations.

Examples of the resulting raw images, converted from ADUs into counts are reported in Fig. 4.

## 2.3 Target selection

### 2.3.1 Selection of candidate stars

We select our candidate stars from the Ariel candidate sample (B. Edwards et al. 2019), which includes a diverse selection of planetary systems that meet Ariel's scientific requirements. From this pool, candidates are chosen based on their brightness levels. Here, we assumed the transit event is always captured in full, and there is only a single transit event happening each time. To adequately sample the signal from the star during the ingress and egress of the planet crossing event (transit), which typically lasts a few minutes, the exposure time must be short, typically less than 45 s. As this part of the challenge of this exercise is to detrend the pointing jitter, we aimed to further shorten the integration times. The jitter timeline is sampled at 1 kHz, meaning that in 45 s, we would combine 45 000 positions, resulting in a blurred image that is impossible to detrend as the spatial information on the focal plane has been averaged out. We set an integration time limit of an order of magnitude smaller than that: 4.5 s, corresponding to 450 jitter positions superimposed. To maintain consistency and comparability within the data set, a fixed exposure time is set for all stellar targets, guiding the selection towards stars with brightness levels compatible with this observing strategy.

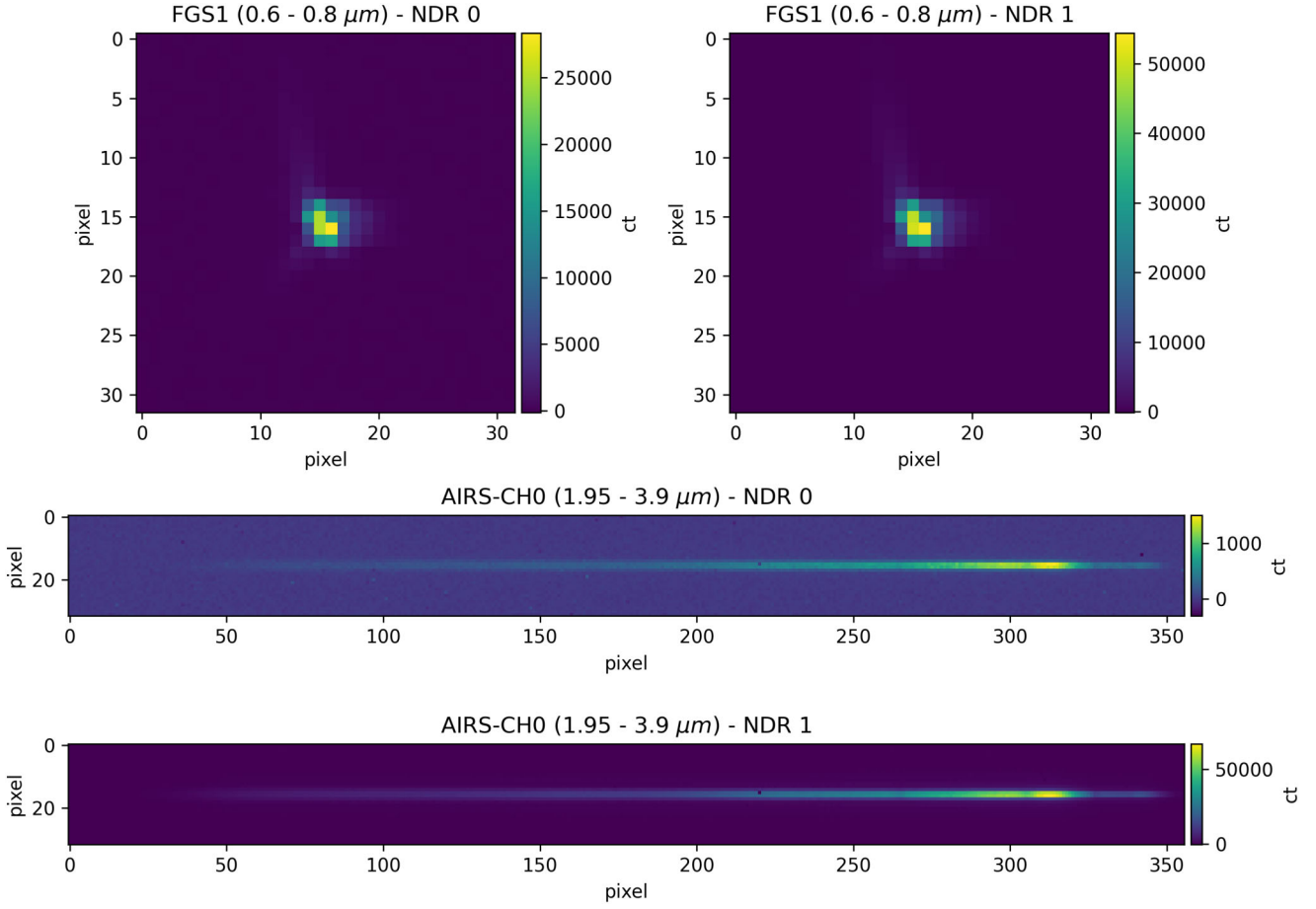
To ensure a high signal-to-noise ratio (SNR), the stars must be sufficiently bright to fill the detector with photons in such a short time, ensuring a photon-noise dominated observation, but not so bright as to saturate the detector.

To satisfy the stringent requirement for integration time while maintaining sufficient SNR, we ended up basing our simulation around four host stars, as reported in Table 1.

Assuming detector saturation and not accounting for the duty cycle, we can estimate the relative uncertainties on the star signal given the saturation times by inverting the signal-to-noise ratio ( $\frac{S}{N}$ ) equation:

$$\frac{S}{N} = \sqrt{\frac{T_{\text{obs}} \cdot n_{e^-}}{\Delta t}}, \quad (5)$$

where  $T_{\text{obs}}$  is the observing time,  $\Delta t$  is the integration time, defined in equation (3), and  $n_{e^-}$  is the number of electrons. Considering the detector saturation levels, we assume  $n_{e^-} = 100 \text{ ke}^-$  for FGS1 and  $n_{e^-} = 85 \text{ ke}^-$  for AIRS-CH0. Under the assumption of saturation in the previously described CDS observation, we adopt a frame time of  $\Delta t = 0.2 \text{ s}$  for FGS1, and  $\Delta t = 4.6 \text{ s}$  for AIRS-CH0. For a total



**Figure 4.** The figure shows the first two NDRs for every channel. NDR 0 is collected at the beginning of the photon collection ramp, while NDR 1 is collected at the end. The figures are reported in counts units as they have been converted from ADUs by inverting equation (4) as  $S_{\text{meas}} = \frac{S_{\text{adu}}}{\text{ADC}_{\text{gain}}} + \text{ADC}_{\text{offset}}$ . The top row shows the photometer FGS1 images, while the bottom row shows the spectrometer AIRS-CH0 images.

**Table 1.** Parameters of the selected sources and models used as input for ExoSim2. Parameters are selected to match B. Edwards et al. (2019).

| Name      | $R_{\star}$ [ $R_{\odot}$ ] | $M_{\star}$ [ $M_{\odot}$ ] | $T_{\text{eff}}$ [K] | $D$ [pc] | Type  | Phoenix model                            |
|-----------|-----------------------------|-----------------------------|----------------------|----------|-------|------------------------------------------|
| KELT-11   | 2.705                       | 1.439                       | 5372.5               | 98       | G8/K0 | lte054.0-3.5-0.0a+0.0.BT-Settl.spec.fits |
| HD 17156  | 1.522                       | 1.32                        | 6066                 | 77.16    | G0    | lte061.0-4.0-0.0a+0.0.BT-Settl.spec.fits |
| HD 209458 | 1.17967                     | 1.1753                      | 6086                 | 47.45    | F9    | lte061.0-4.5-0.0a+0.0.BT-Settl.spec.fits |
| HD 149026 | 1.468                       | 1.31                        | 6158                 | 78       | G0    | lte062.0-4.0-0.0a+0.0.BT-Settl.spec.fits |

**Table 2.** Orbital parameters used for HD209458 b as reported in K. G. Stassun, K. A. Collins & B. S. Gaudi (2017). The mid-transit time has been set in this work to align the transit time with the simulated observation.

| Parameter                  | Value     |
|----------------------------|-----------|
| mid-transit time ( $t_0$ ) | 3.75 h    |
| period ( $p$ )             | 3.525 d   |
| semi-major-axis ( $a$ )    | 8.81      |
| inclination ( $i$ )        | 86.71 deg |

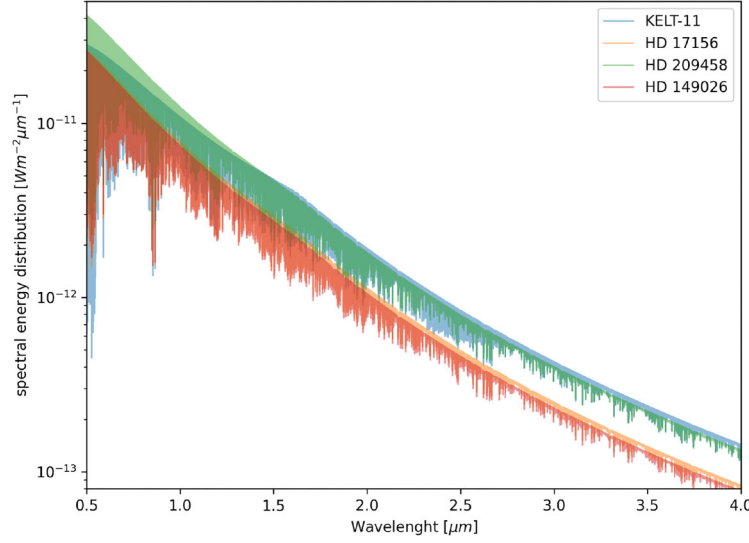
observing time of  $T_{\text{obs}} = 7.5$  h, the relative noise can then be estimated as

$$\left(\frac{N}{S}\right)_{\text{FGS1}} = \sqrt{\frac{0.2 \text{ s}}{7.5 \text{ h} \cdot 100 \text{ ke}^-}} = 8 \text{ ppm.} \quad (6)$$



**Table 3.** Specification of different train test scenarios, including the  $R_p$  range used to uniformly sample planet radii, host stars used as the sources of flux, trace gases included in the atmospheres, and different detector types and noise maps for the instrument.  $\mathcal{U}(x_1, x_2)$  represents a uniform distribution between  $x_1$  and  $x_2$ .

|                                                      | Training                                                    | Training-Ext                                                | Test case 0                                                             | Test case 1                                                             | Test case 2                                                             | Test case 3                                                             | Test case 4                       | Test case 5                         | Calib case    |
|------------------------------------------------------|-------------------------------------------------------------|-------------------------------------------------------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------|-----------------------------------|-------------------------------------|---------------|
| <b>Number of data</b>                                | 576                                                         | 97                                                          | 104                                                                     | 102                                                                     | 103                                                                     | 105                                                                     | 105                               | 231                                 | 4             |
| <b><math>R_p</math> range (<math>R_{Jup}</math>)</b> | 0.5–1.2                                                     | 0.5–1.2                                                     | 0.5–1.2                                                                 | 0.5–1.2                                                                 | 0.5–1.2                                                                 | 0.2–1.5                                                                 | 0.5–1.2                           | 0.5–1.2                             | Actual planet |
| <b>Host Stars</b>                                    | KELT-11, HD 17194                                           | KELT-11, HD 17194                                           | KELT 11, HD 17194                                                       | KELT-11, HD 17194                                                       | KELT-11, HD 17194                                                       | HD 209458, HD 149026                                                    | KELT-11, HD 17194                 | KELT-11, HD 17194                   | All four      |
| <b>Trace gases</b>                                   | H <sub>2</sub> O, CH <sub>4</sub> , CO <sub>2</sub> , Cloud | H <sub>2</sub> O, CH <sub>4</sub> , CO <sub>2</sub> , Cloud | NH <sub>3</sub> , C <sub>2</sub> H <sub>2</sub> , SO <sub>2</sub> , SiO | NH <sub>3</sub> , C <sub>2</sub> H <sub>2</sub> , SO <sub>2</sub> , SiO | NH <sub>3</sub> , C <sub>2</sub> H <sub>2</sub> , SO <sub>2</sub> , SiO | NH <sub>3</sub> , C <sub>2</sub> H <sub>2</sub> , SO <sub>2</sub> , SiO | HCN, H <sub>2</sub> S, AlH and CO | HCN, H <sub>2</sub> S, AlH and CO   | flat          |
| <b>Detector type</b>                                 | Model A                                                     | Model A                                                     | Model A                                                                 | Model B                                                                 | Model C                                                                 | Model A                                                                 | Model A                           | Model A                             | Model A       |
| <b>Semi-Major Axis</b>                               | Fixed                                                       | $\mathcal{U}(8.81, 11)$                                     | Fixed                                                                   | Fixed                                                                   | Fixed                                                                   | Fixed                                                                   | Fixed                             | $\mathcal{U}(8.81, 15.86)$          | Fixed         |
| <b>Inclination</b>                                   | Fixed                                                       | $\mathcal{U}(86.71, 88)$                                    | Fixed                                                                   | Fixed                                                                   | Fixed                                                                   | Fixed                                                                   | Fixed                             | $\mathcal{U}(87, 90)$               | Fixed         |
| <b>Mid Transit Time</b>                              | Fixed                                                       | $\mathcal{U}(t_0 - 0.1, t_0 + 0.1)$                         | Fixed                                                                   | Fixed                                                                   | Fixed                                                                   | Fixed                                                                   | Fixed                             | $\mathcal{U}(t_0 - 0.5, t_0 + 0.5)$ | Fixed         |



**Figure 5.** Stellar spectral energy distribution (SED) used in this work for each star used as target and reported in Table 1.

For FGS1 the duty cycle plays a significant role. Since only 40 per cent of the total 7.5 h is effectively used for integration, the relative noise increases by a factor of  $1/\sqrt{0.4}$ . Therefore, the expected value is

$$\left(\frac{N}{S}\right)_{\text{FGS1,DC}} = \frac{8}{\sqrt{0.4}} = 12 \text{ ppm.} \quad (7)$$

In contrast, for AIRS the duty cycle can be neglected, and we estimate

$$\left(\frac{N}{S}\right)_{\text{AIRS}} = \sqrt{\frac{4.6 \text{ s}}{7.5 \text{ h} \cdot 85 \text{ ke}^-}} = 45 \text{ ppm.} \quad (8)$$

These values represent the theoretical lower limit for the expected uncertainties for the selected stars, assuming they saturate the detector and utilize the full instrument capability.

In real observations, each star would have a different exposure time, leading to variations in data set sizes. This tailored approach ensures that the observations are optimized for both the instrument and the target star characteristics. However, for this exercise, we standardized the integration time and the observing time to 7.5 h. We ensured that every planet has the same transit time, and the mid-time of the simulated observation coincides with or is close to the transit event mid-time. While we realize this standardized integration and observing time setup is not necessarily representative of the normal operating conditions, the homogeneity in the data sets allows for easier introduction to ML-related approaches.

Nevertheless, the chosen observing duration remains physically realistic within the context of the Ariel mission. Ariel is expected to observe each target for a total duration of approximately 2.5 times the transit duration, in order to adequately sample both in-transit and out-of-transit baselines. Therefore, a total observing time of 7.5 h corresponds to a transit duration of approximately 3 h, which is representative of typical hot Jupiter systems and consistent with the class of targets considered in this work. This value is also broadly consistent with the average transit duration of the exoplanet population targeted by Ariel, ensuring that the simulated observations reflect realistic temporal scales. This choice ensures that, despite the imposed uniformity, the simulated observations retain a realistic temporal structure and signal-to-noise regime.

### 2.3.2 Stellar models

To simulate the incoming light from the star, we use the stellar spectral energy distribution (SED) obtained from a grid of synthetic Phoenix spectra<sup>10</sup> (I. Baraffe et al. 2015). The best matching model is selected based on stellar temperature, surface gravity (logg), and metallicity. The specific model for each star is chosen from the indicated SED grid according to these parameters, scaled by  $(R_*/D)^2$  to account for the star distance and is reported in Table 1. The resulting SED has units of  $\text{W} (\text{m}^2 \cdot \mu\text{m})^{-1}$  and it is reported in Fig. 5.

<sup>10</sup>[https://phoenix.ens-lyon.fr/Grids/BT-Settl/CIFIST2011\\_2015/FITS/](https://phoenix.ens-lyon.fr/Grids/BT-Settl/CIFIST2011_2015/FITS/)

**Table 4.** Abundance range for sampling different physical parameters and their corresponding line list references.

| Parameters                    | Abundance range<br>(log <sub>10</sub> ) | Line list reference                                   |
|-------------------------------|-----------------------------------------|-------------------------------------------------------|
| H <sub>2</sub> O              | -8 to -4                                | O. L. Polyansky et al. (2018)                         |
| CO <sub>2</sub>               | -8 to -4                                | L. S. Rothman et al. (2010)                           |
| CH <sub>4</sub>               | -8 to -4                                | S. N. Yurchenko et al. (2017)                         |
| NH <sub>3</sub>               | -8 to -4                                | P. A. Coles, S. N. Yurchenko<br>& J. Tennyson (2019)  |
| C <sub>2</sub> H <sub>2</sub> | -8 to -4                                | K. L. Chubb, J. Tennyson &<br>S. N. Yurchenko (2020)  |
| SO <sub>2</sub>               | -8 to -4                                | D. S. Underwood et al. (2016)                         |
| SiO                           | -8 to -4                                | J. Buldyreva, S. N. Yurchenko<br>& J. Tennyson (2022) |
| HCN                           | -8 to -4                                | R. J. Barber et al. (2014)                            |
| H <sub>2</sub> S              | -8 to -4                                | A. A. A. Azzam et al. (2016)                          |
| AlH                           | -8 to -4                                | S. N. Yurchenko et al. (2024)                         |
| CO                            | -8 to -4                                | G. Li et al. (2015)                                   |
| grey cloud (Pa)               | 3 to 6                                  | N/A                                                   |

## 2.4 Planetary model

This section describes the planetary models used in data set generation. To ensure diverse representation of the planetary population, we varied the following categories of parameters: planetary configurations ( $R_p$ , inclination, semimajor axis, mid-transit time) and atmospheric features (equilibrium temperature, trace gases detailed in Table 4, and clouds) during the simulation process. We note that these variations cover hotter and larger exoplanets to enrich the diversity. Table 3 provides a detailed breakdown of varied parameters; below we describe the planetary configurations and atmospheric models in detail.

### 2.4.1 Planetary configuration

The planetary configuration concerns with the specifics of the planetary system.

The configuration of the host star follows the targets chosen in Table 1. We opt to build examples using synthetic data sets, i.e. synthetic planets orbiting around our selected host stars. We sampled the radii of different synthetic planets from a uniform distribution and used a Splined-based mass radius predictor to infer the corresponding mass (Yip et al., in preparation) from the uniformly sampled radius. We have set a lower bound in the planet's mass as 10 Earth Masses in this investigation. As part of the effort to maintain a homogeneous data set, we have assumed these planets to resemble very similar orbital configurations as HD209458 b, as described in Table 2. This assumption is relaxed in some cases, as indicated in Table 3. Fixed settings mean the configuration for HD209458 b is used, otherwise, they are varied according to a specified uniform distribution. This is to allow more extensive tests on the algorithm based on possible scenarios that could occur during an observation.

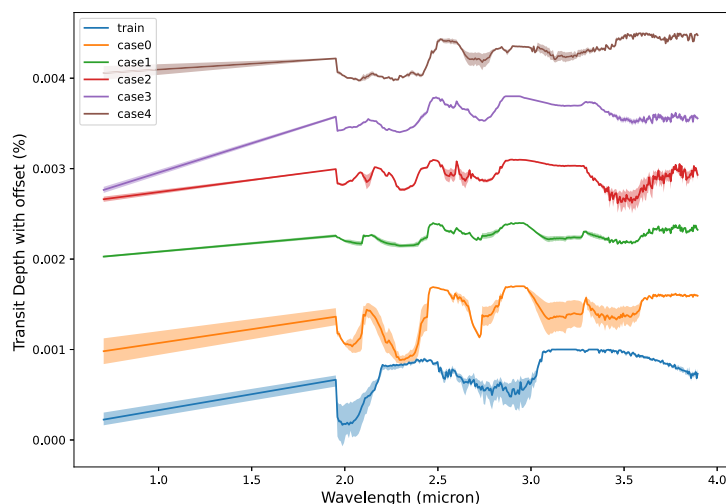
### 2.4.2 Atmospheric model

To generate realistic simulations of observations from distant exoplanets and their atmospheres, we employed the forward modelling capabilities of TauREx3. This tool allowed us to create hypothetical atmospheric modulations for these observations.

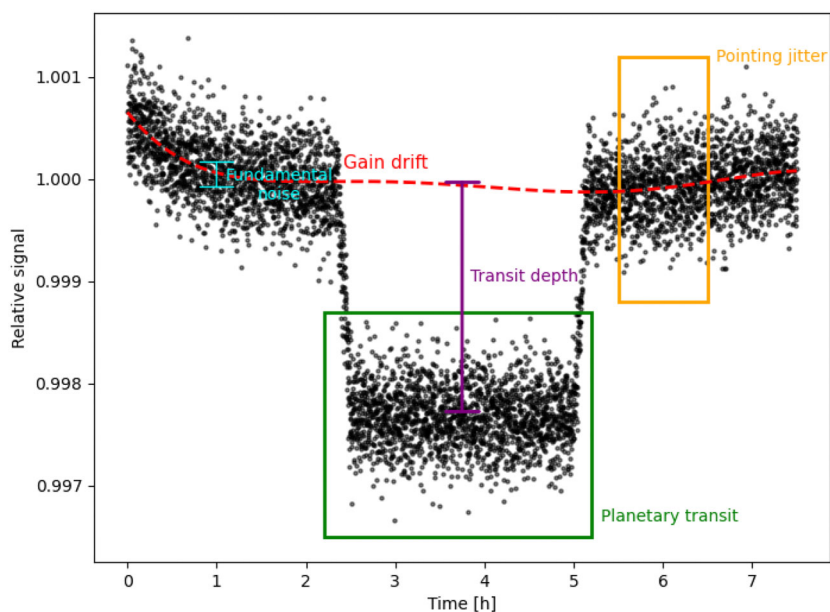
Within reasonable assumptions for our synthetic planets, and with the goal of providing a benchmark data set for data detrending, we set up a number of 'global' assumptions for our atmospheric model: we created all the atmospheres with a 1D plane-parallel primary atmosphere composed of H/He at a solar abundance ratio of 0.17, along with additional atmospheric absorption such as Collision-Induced-Absorption (C.I.A.) and scattering effects such as Rayleigh Scattering. Furthermore, we assumed an isothermal Temperature-Pressure profile. We used the equilibrium temperature of the planet as the temperature of the planet's atmosphere. The equilibrium temperature is calculated via the following relationship from G. Tinetti, T. Encrenaz & A. Coustenis (2013):

$$T_{\text{eqm}} = T_{\star} \times \sqrt{\frac{R_{\star}}{2a}} \times (1 - A_B)^{1/4}, \quad (9)$$

where we have assumed the Bond albedo ( $A_B$ ) to be 0, and that the host stars' radiation follows a blackbody. The exact values of  $T_{\star}$  and  $R_{\star}$  are varied according to their measured uncertainty (see Table 1). The extent of the atmosphere is set from  $10^{-6}$  to  $10^6$  Pa. All observation is conducted in transmission mode and is generated in native resolution as provided from the ExoMol linelist data with no binning applied (see relevant references in Table 4). In some cases, we implemented a grey cloud deck with pressures varying from 3 to 6 bar. We note here that the level of variation does not fully capture current findings for hot Jupiters, where high-altitude clouds could exist at lower pressures; this limitation will be addressed in future editions of the data set.



**Figure 6.** Visualization in spectral space of the different training and test cases presented in Table 3. For each spectrum, we show the median-subtracted spectrum for that training or test case and the spread of the spectra to show the shape. Offsets are applied to individual spectra for better visualization.



**Figure 7.** White light curve (i.e. the resultant light curve from using the spectrometer as a single wavelength channel) for one of the simulated observations with Ariel AIRS-CH0. The signal dip from the transit event is highlighted in the green box, with the depth of the transit also marked in purple. We highlighted other significant features: the spread of the measurements due to high-frequency and quantum noise (cyan), the polynomial drift due to simulated gain variations (red), and the photometric noise caused by the jitter disturbance affecting the measurements (orange), mostly embedded in the overall noise in this case.

Aside from these global assumptions, trace gases of various abundances are added to the atmosphere depending on the different scenarios. All of them are assumed to have constant chemical abundance throughout the extent of the atmosphere. Please see Table 3 for the exact species used in each case and Fig. 6 for a comparison of each case in spectral space. We note that some of these combinations and prior ranges may not be physically motivated and do not fully capture the diversity of targets that Ariel is going to observe. These settings are set up to prevent any algorithms from learning to detrend by having prior assumptions on the spectral feature of particular molecules or combinations of molecules (in other words, having prior knowledge before the extraction), and to stress test any potential models with their ability to detrend accurately and precisely, even in unknown and unseen scenarios. On top of that, we have also discarded spectra with NaN values and with transit depth larger than 2 per cent.

## 2.5 Data production and sanitation

For each simulated observation, we produced three versions of the same data set. (a) The first version includes all noise sources, representing a complex data set on which the pipelines can be trained and tested under realistic conditions. (b) A second version excludes



**Table 5.** Data set information for the two detectors.

| Parameter                 | FGS 1   | AIRS-CH0 |
|---------------------------|---------|----------|
| Observing time            | 7.5 h   | 7.5 h    |
| Integration time          | 0.2 s   | 4.6 s    |
| Number of images per obs. | 135 000 | 11 250   |
| Number of CDS couples     | 67 500  | 5625     |
| Size of image             | 32 × 32 | 32 × 356 |

time-correlated noise components, such as jitter and gain drift, which require dedicated analysis strategies. Finally, (c) a third version is generated without any noise or uncertainty, serving as a reference data set.

To generate these data sets, we developed a distributed computing system that operates across multiple machines within the Department of Physics and Astronomy at Cardiff University. This distributed approach allowed us to produce the entire data set (over 7 TB, as detailed later) in less than a day of computing time.

Since ExoSim2 is designed to model the effects of various instrumental subsystems in detail, its output includes all intermediate products required to reconstruct the simulated observation. It was therefore necessary to sanitize the data to provide only the essential versions. Specifically, we removed all auxiliary data, retaining only the data cube with the simulated images and the critical information necessary for their interpretation (wavelength solution, time axis, and integration times for each image). In addition, we supplied a separate set of calibration data with parameters uniformly altered by 5 per cent to simulate realistic uncertainties in calibration products, rather than providing the exact inputs used in the data generation process.

### 2.5.1 Data set statistics

The complete data set generated for this exercise comprises 7.0 TB and is stored on the Cardiff University and DiRAC CSD3 server, accessible upon request. This data set includes 1435 simulated observations, as described in Section 2.5. Each simulated observation weighs approximately 4.5 GB, as it contains all intermediate products of the simulation.

A sanitized version of the training data set, publicly available, consists of 673 observations with a total size of approximately 174.33 GB (with each observation in the sanitized version reduced to about 500 MB). This public data set includes images, calibration data, and ground truth spectra, as detailed in Section 2.6. All observations were simulated using the same integration and observation times to ensure uniformity. Each data set contains 135 000 images for FGS1 and 11 250 images for AIRS-CH0. The test data set, while hidden on Kaggle, will be made available on Zenodo, it has a total size of 184 GB with 758 files

Each pair of images represents the extremes Non-Destructive Readouts (NDRs) in a CDS readout mode for the detector ramp. These images are separated by 0.2 s for FGS1 and 4.6 s for AIRS-CH0, resulting in 67500 CDS pairs for FGS1 and 5625 CDS pairs for AIRS-CH0. The total observing time across all observations is 7.5 h.

A summary of this information, including the number of observations, images, and the total observing time, is provided in Table 5.

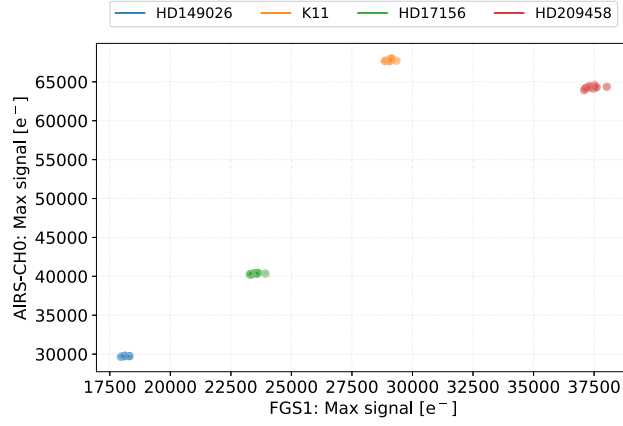
## 2.6 Data validation and overview

ExoSim2 provides a detailed time-domain model of both the payload and astrophysical scene, enabling accurate simulation of Ariel's observational data with realistic noise and systematic effects (Fig. 7). The simulator has undergone rigorous validation, with each module tested to achieve over 90 per cent test coverage, ensuring robust and reliable outputs. Given the substantial scale of the generated data set, comprehensive end-to-end validation becomes critical for confirming input accuracy and minimizing potential errors. This section examines the validation methodology and presents an overview of the resulting data set, demonstrating its value for the Data Challenge while highlighting opportunities for broader scientific exploration.

### 2.6.1 Data processing pipeline

The data processing pipeline was developed to accurately extract transit signals, evaluate noise properties, and validate the data set against theoretical models. The pipeline proposed here is a basic implementation that does not reflect the official Ariel pipeline, which is still under development at the time of writing. The methodology can be divided into three main phases: pre-processing, image construction, and spectral analysis.

In the pre-processing phase, the quality of the raw data was ensured by identifying and masking defective pixels, such as dead or hot pixels. This was achieved using calibration data and the `sigma_clip` method provided by the `astropy` package. Following this, the raw 16-bit digital data were converted into 64-bit floating-point values. During this conversion, corrections for the analogue-to-digital gain ( $\text{ADConv}_{\text{gain}}$ ) and offset ( $\text{ADConv}_{\text{offset}}$ ) were applied using the calibration files (see Section 2.2.6 and equation 4). Pixel non-linearity, a common feature of detector response, was then corrected using a fifth-order polynomial fit derived from calibration



**Figure 8.** Maximum signal detected out-of-transit for the photometer (FGS1) and spectrometer (AIRS-CH0). The colour code indicates the stellar targets, illustrating how differences in stellar spectral energy distributions influence the observed signals.

coefficients (see Section 2.2.2 and equation 2). Finally, the contribution of dark current was subtracted for each pixel, scaled appropriately by the integration time, using detector-specific calibration maps (see equation 1).

The second phase, image construction, involved creating science frames optimised for analysis. First, CDS frames were computed by subtracting the signal from the first and last NDRs within each integration ramp (see equation 3). Subsequently, pixel-to-pixel response variations were corrected by normalizing the CDS frames with flat-field coefficients provided in the calibration data. This step ensured a uniform response across the detector array. The signals were then integrated across the images to obtain transit light curves. For photometric observations (FGS1), the total flux was summed over the defined aperture, while for spectroscopic observations (AIRS-CH0), the signal was integrated along pixel columns and associated with specific wavelengths. Any data outside the target wavelength range of 1.95–3.9  $\mu\text{m}$  was excluded to maintain spectral consistency.

The final phase of the data processing pipeline focused on extracting transit depths and estimating noise properties from the transit light curves obtained in the last step. To isolate the transit signal, the timeline was divided into transit and out-of-transit regions. This separation leveraged the symmetry of the transit event or, when necessary, an estimation of the timeline mid-point. Margins were applied around the ingress ( $T_1$ ) and egress ( $T_4$ ) to account for potential edge effects. The transit depth was then calculated using the relationship:

$$d(t, \lambda) = 1 - \frac{F_{\text{in}}}{F_{\text{out}}}, \quad (10)$$

where  $F_{\text{in}}$  and  $F_{\text{out}}$  represent the stellar flux rates ( $F(\lambda)$ , in  $\text{e}^- \text{s}^{-1}$ ) measured during and outside the transit, respectively. To first order, the transit depth can be approximated as  $d(t, \lambda) \simeq (R_p/R_*)^2$  (see e.g. D. K. Sing 2018), linking the depth to the relative sizes of the planet and the star. The out-of-transit timelines were then detrended by fitting a fifth-order polynomial, removing systematic gain drifts.

The expected uncertainty on the transit depth, derived from equation (10), is expressed as:

$$\sigma_d^2 = \left( \frac{1}{t_{\text{in}} F_{\text{out}}} \right)^2 v_{\text{in}} + \left( \frac{F_{\text{in}}}{t_{\text{out}} F_{\text{out}}^2} \right)^2 v_{\text{out}}, \quad (11)$$

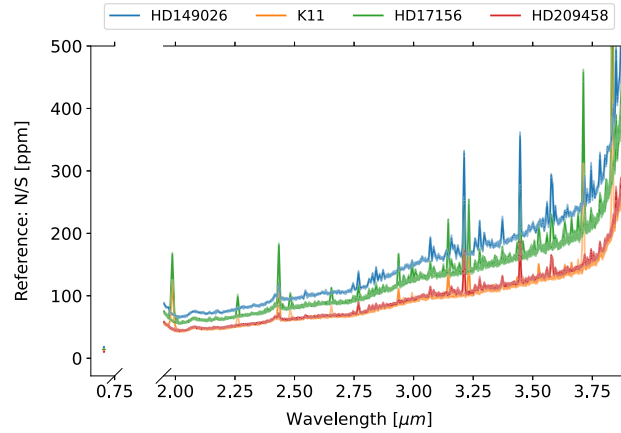
where  $t_{\text{in}}$  and  $t_{\text{out}}$  are the times spent in and out of transit and  $v_{\text{in}}$  and  $v_{\text{out}}$  represent the time-integrated variances in and out of transit. These variances are estimated from the residuals between the measured light curves and the ground truth transit models.

This simple prototype pipeline, although not designed for comprehensive data reduction, successfully isolates planetary transit signals and quantifies measurement uncertainties, validating the reliability of the data set.

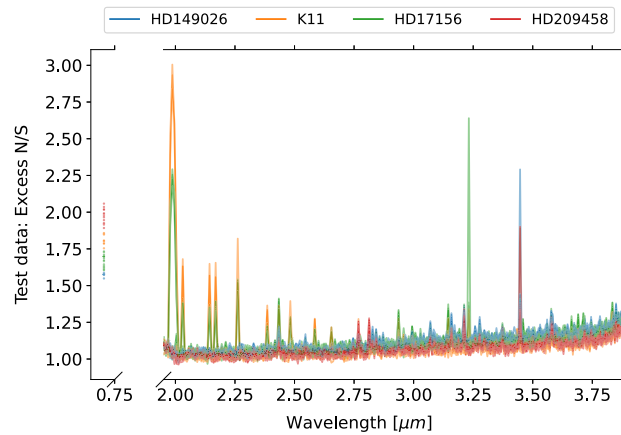
### 2.6.2 Sample validation

This section details the validation performed on a subset of simulated observations to evaluate signal and noise levels. For this analysis, we selected a subsample of the data set comprising 100 examples. These examples have the largest peak-to-valley measured across their ground truth spectra than all the other examples in the data set. This initial sample was further pruned to include only those observations where the transit mid-point was centred within the timeline. The final subsample consisted of 46 simulated observations.

The first step of this validation was to compute the maximum signal in each channel. This was done by taking the median value along the time axis of the out-of-transit CDS science frames and identifying the pixel with the highest signal. The results, presented in Fig. 8, show that the signal spans from 17.6  $\text{ke}^-$  to 37.5  $\text{ke}^-$  in FGS1 and from 30  $\text{ke}^-$  to 68  $\text{ke}^-$  in AIRS-CH0. These ranges probe different depths of the detector well in both channels while avoiding saturation. Differences in the stellar SEDs explain why KELT-11



**Figure 9.** Relative noise (ppm) as a function of wavelength for the reference timelines. The colour coding represents different stellar targets, reflecting how their SEDs influence the noise levels across the spectrum. The large spikes come from detector imperfections such as bad pixels that are left uncorrected.



**Figure 10.** Excess relative noise (ppm) in the test data compared to the reference timelines, plotted as a function of wavelength. Colour coding corresponds to stellar targets, highlighting trends due to line-of-sight jitter and noise dominance.

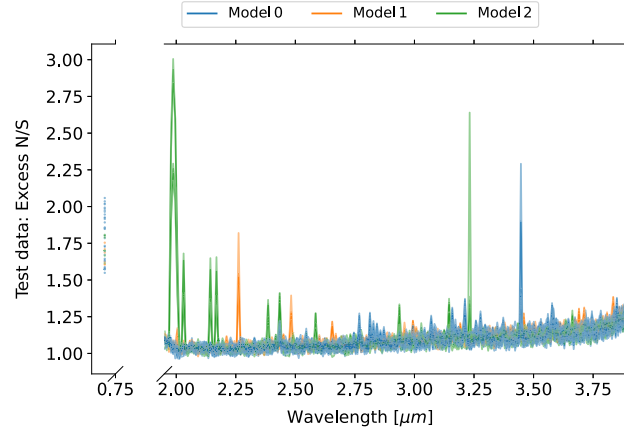
exhibits the highest signal in AIRS-CH0 with respect to the signal measured in AIRS-CH0 for the other stars. Similarly, the HD 209458 signal measured in FGS1 dominates with respect to the FGS1 signal obtained for the rest of the stellar sample. Overall, we observe that for each of these stars, the AIRS-CH0 signal is stronger than that in FGS1, as expected.

From here, the validation strategy is to use two different versions of the data set (introduced in Section 2.5) for a self-consistent and homogeneous comparison of their corresponding results, with utmost control on the simulation inputs. Specifically, we compare the data set version including all noise sources (version **a**; hereafter, ‘test data’) with that containing only fundamental noise sources (version **c**; hereafter, ‘reference’).

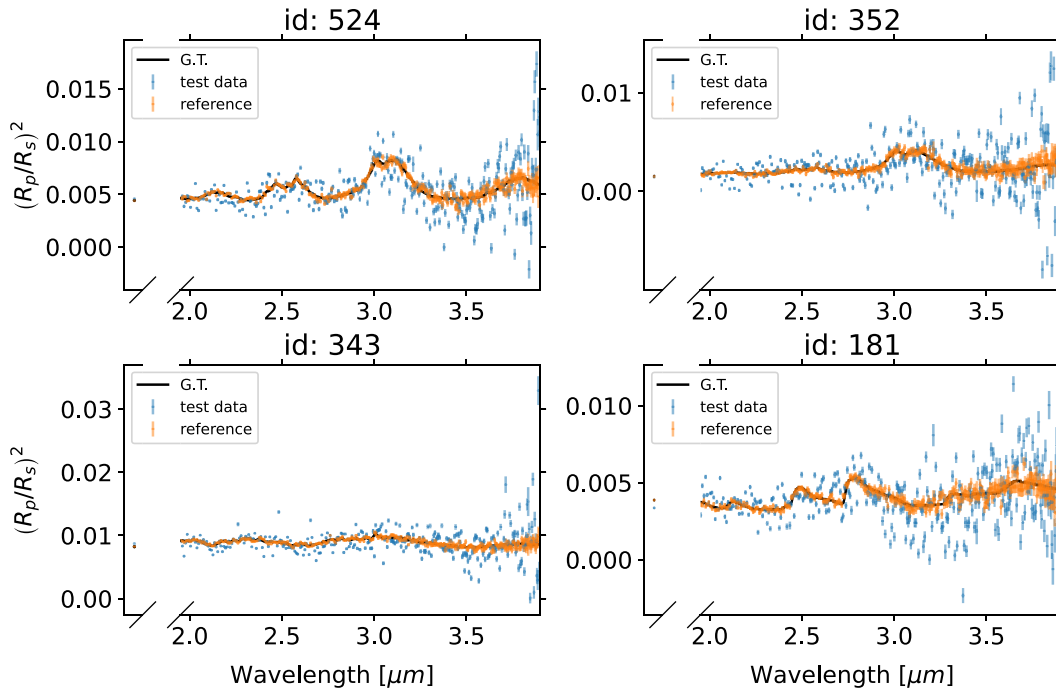
Fig. 9 illustrates the relative noise as a function of wavelength, calculated from the out-of-transit variance in the timelines and rescaled to the observation duration. Here, we use the reference timelines. Differences between stars arise from their distinct SEDs. Differences in the curves with the same star arise from variations in the simulated detector, as we show in the following. Spectral features below the noise curves in this figure are effectively undetectable, even assuming ideal data reduction.

Then, we evaluated the excess relative noise in the test data, by comparing it to the reference timelines. In this way, any excess noise measured in the test data with respect to the reference data can be traced to noise sources that are not corrected or insufficiently corrected, e.g. from correlated and time-dependent systematics. The noise excess, when present, can translate to biases in the measured transit spectra, hindering our interpretation of atmospheric results.

Fig. 10 shows that in FGS1, excess noise increases with the signal level due to uncorrected line-of-sight jitter dominating over fundamental noise sources. This trend is also visible in the brightest spectral bins of AIRS-CH0 at the blue edge of the band but reverses for wavelengths longer than 2.2  $\mu\text{m}$ , where photon noise and readout noise dominate. Detector-specific effects, such as bad pixels, result in the additional noise spikes shown in the figure. This is evidenced in Fig. 11, where the colour coding represents the different detector models used in the simulations. Each detector model has bad pixels appearing at different positions along the spectral trace, which cause an increase in the measured noise in the corresponding wavelength bins when no bad pixel interpolation is applied.



**Figure 11.** Excess relative noise (ppm) as in Fig. 10, but with colour coding representing each of the different detector models used in the data creation.



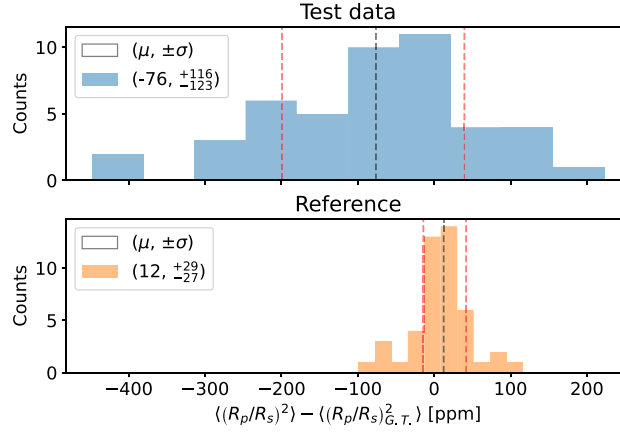
**Figure 12.** Comparison of transit spectra for selected examples from the subsample. Black: ground truth spectra; orange: reference data; blue: test data. Error bars are included.

Next, we assessed the accuracy of the extracted transit depths by comparing test and reference spectra to the ground truth. Fig. 12 shows examples of extracted spectra, where the reference timelines align well with the ground truth, with  $68.6^{+3.0}_{-2.7}$  per cent of points within the  $1\text{-}\sigma$  interval. For the test data, only  $17.0^{+3.5}_{-2.1}$  per cent fall within this range, indicating the limitations of the simplistic reduction applied. Fig. 13 highlights a bias toward smaller transit depths in the test data ( $\sim 100$  ppm), with residuals between extracted and ground truth spectra showing larger deviations.

While the data reduction used here does not fully address systematic uncertainties, it provides a useful baseline for participants in the Data Challenge and other follow up studies. The validation confirms that the signals and noise levels in the data set align with input parameters, demonstrating its suitability for benchmarking future analysis pipelines.

### 3 DATA PARTITION

A unique challenge in exoplanet characterization lies in the absence of ground truth – for most cases, we do not know a priori the kind of atmosphere the telescope will observe. An ideal data reduction pipeline for exoplanetary data, or space-based data by extension, should be able to retrieve the signal without bias within the permitted noise level (above or close to the photon noise level).



**Figure 13.** Histogram of residuals between median extracted transit depths and ground truth. The black line shows the mean and the red dash line shows the standard deviation. *Top panel:* test data, showing larger deviations and a negative bias. *Bottom panel:* reference data, with deviations within  $\pm 100$  ppm.

To thoroughly test any pipeline – whether parametric, machine learning, or hybrid – we have designed a data partition that explicitly includes out-of-distribution test examples, as detailed in Table 3. Unless otherwise stated, the global atmospheric setup follows the discussion in Section 2.4. This partitioning strategy divides the data set into training (including validation) and multiple test scenarios that systematically introduce domain shifts across key observational parameters. Each test case is designed to probe a specific type of distribution shift that algorithms will encounter when deployed on real Ariel observations, where target properties will inevitably differ from any training set.

The training set establishes a baseline with simulated planets having radii between 0.5 and  $1.2 R_{\text{Jupiter}}$  orbiting two host stars (KELT-11 and HD 17194) with atmospheric compositions including  $\text{H}_2\text{O}$ ,  $\text{CH}_4$ ,  $\text{CO}_2$ , and clouds, based on currently discovered molecular species. The Training-Ext set is an extended training set with uniform distributions for semimajor axis, inclination, and mid-transit time to introduce temporal and geometric variability in the light curves, consistent with likely scenarios in actual observations.

The test scenarios progressively introduce distribution shifts to explicitly test whether models overfit to the training set. Test cases 0–2 maintain the same atmospheric species and host stars as the training set but employ different detector models (A, B, C), which differ in their bad pixel distributions, flat-field characteristics, and noise maps (see Section 2.2.1). This configuration isolates instrumental robustness: can algorithms handle detector variations while the astrophysical signal remains constant? Test case 3 expands to additional host stars (HD 209458 and HD 149026) with different effective temperatures and SEDs compared to the training stars, with a broader planetary radius range ( $0.2\text{--}1.5 R_{\text{Jup}}$ ) that includes smaller planets absent from training. This tests generalization to different stellar irradiation environments and planetary bulk properties. To implement thorough screening of proposed pipelines, Test cases 4–5 feature entirely different atmospheric compositions (HCN,  $\text{H}_2\text{S}$ , AlH, and CO) from those in the training set and Test cases 0–2, producing distinct spectral morphologies. Test case 5 extends Test case 4 with additional orbital parameter variations (semimajor axis, inclination, mid-transit time) similar to Training-Ext, introducing temporal and geometric diversity alongside compositional shifts. This stringent combination tests whether algorithms have learned general denoising principles or have simply memorized training set spectral shapes. The calibration case uses flat (featureless) spectra with no atmospheric features for all four stars, representing the null hypothesis and providing a baseline for assessing systematic biases in the extraction process. Note that these four calibration cases are not included in the actual test data in the Kaggle competition, but are provided for independent validation.

We emphasize that test case design prioritizes algorithmic stress-testing over strict astrophysical plausibility. While some atmospheric combinations may not represent the most probable Ariel targets, they serve the essential function of exposing overfitting and validating that pipelines can handle diverse, unexpected scenarios – a critical requirement for survey missions. This partitioning approach deliberately exposes models to the types of challenges they will encounter when analysing real Ariel observations, where target exoplanets may have atmospheric compositions, stellar hosts, and orbital configurations that differ significantly from any training examples.

#### 4 ML-BASED DATA REDUCTION PIPELINE

To explore the possibility of having a scalable data reduction pipeline for large data sets, and to demonstrate how to manipulate the data set for data science applications, we developed a ML-based pipeline to predict the combined transmission spectra derived from FGS1 and AIRS-CH0 instruments, or in another words, a 283 long vector. In this section we will outline its implementation. We begin our pipeline with data pre-processing, including calibrating the science frames using the calibration files, time-binning, and normalizing the data set. The next stage is to feed the calibrated and normalized data into the neural network based pipeline. Here we present two different



architectures, each leveraging different perspective of the data. The final stage is using the trained networks to predict planetary spectra and associated uncertainties for the test set, which will be compared against the ground truth for scoring.

#### 4.1 Data pre-processing and binning

We begin the data pre-processing by calibrating the observation. We performed Analogue to Digital Conversion (ADConv), flat fielding, dark current subtraction, dead pixel corrections and non-linearity correction using the calibration files provided for each observation. Each science image is corrected using the following formula:

$$\frac{\text{image} - \text{dark}}{\text{flat}}. \quad (12)$$

In all these frames the dead pixels are masked using Python masked arrays. The bad pixels are thus masked but left uncorrected. Other methods can be used to correct bad-pixels, but we leave this for future exploration. The non-linearity of the pixel-wise response function is corrected using the inverse polynomial coefficients provided in the calibration files as well, to perform non-linearity correction.

Following the assumptions set out in Section 2.2.5, we computed the CDS using the difference between the end of exposure and the start of exposure. This resulted in a data set containing 135 000 CDS frames for FGS1 and 5610 CDS frames for AIRS-CH0.

We subsequently performed time binning on the science frames from both instruments to save storage space and homogenize the input to the neural network. The binned AIRS-CH0 frames were created by grouping 30 CDS frames together, corresponding to 187 frames with a time exposure of 2'24" each. To match the input AIRS-CH0 data, we applied a binning of  $30 \times 12$  (360) on the input FGS1 data. This consisted of 30 frames to match the AIRS-CH0 binning, and 12 frames to match the number of FGS1 images taken for each AIRS-CH0 frame.

The resultant datacubes of frames have dimensions of (187, 32, 32) for FGS1 and (187, 32, 282) for AIRS-CH0. We assume that this binning will have a small impact on the relevant information (noise + signal) contained in the data, as the dominant effect is the non-linearity correction (shown in Appendix in Section A), which is proportional to the signal. We note, however, that the jitter motion may be blurred as a result of the binning step. Additionally, we combined the AIRS and FGS images by summing up the 32 pixels of the FGS x-axis, transforming each image into a (32, 1) format. We then concatenated this with the AIRS-CH0 image, yielding an input dimension of (187, 32, 283) for each combined image. The final step also aligns with the output target spectra, which have 283 points, one for FGS1 and 282 for AIRS-CH0.

Given the predominance of stellar flux in the signal, we implemented a normalization procedure using the stellar spectrum extracted from the images. Fig. 14 shows an example of the signal before and after the normalization has been applied.

#### 4.2 Learning strategy overview

A key feature of the data set is its rich volume in information content, but exceedingly limited amount of training examples, standing from a machine learning point of view. Keeping data close to its original form preserves information like instrumental noise and bias, but increases computation costs and may hinder neural network convergence.

There are two broad learning strategies:

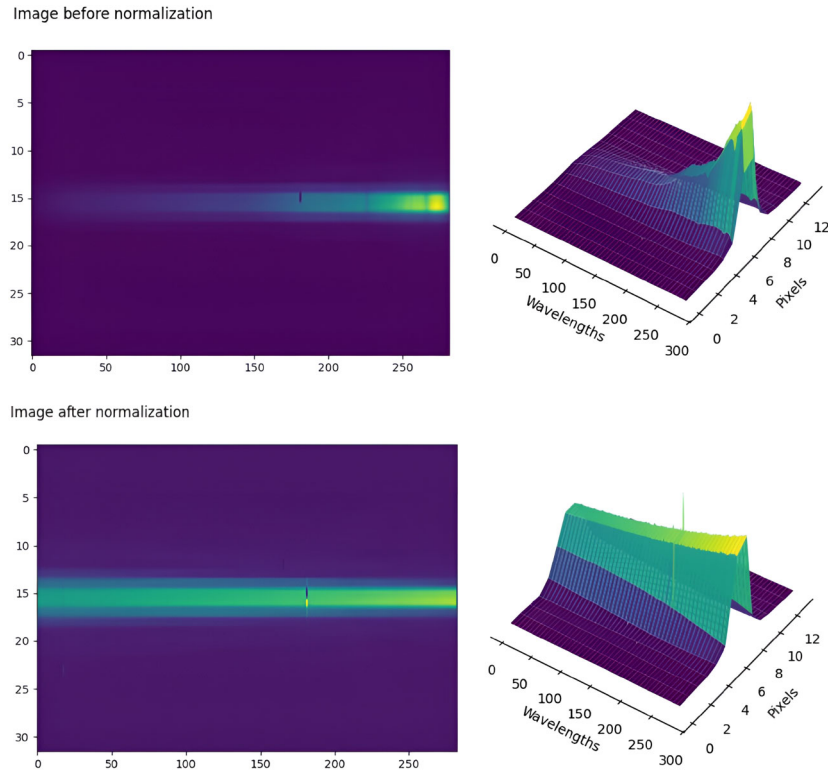
- (i) Training on complete spatial-spectral time-series to generate full spectra.
- (ii) Column-wise analysis along the spectral direction to determine transit depth per column.

The former strategy is straight forward but computationally intensive, while the second strategy facilitates the construction of a substantial data set ( $N_{\text{data}} \times 282$ ). For this investigation, we have chosen to prioritize the first strategy for the following reason: The individual light curves extracted were excessively noisy, affecting our ability to accurately determine the mean spectrum value. Although we could improve the signal-to-noise ratio (SNR) by summing the flux across all wavelengths in each image to generate a 'white' light curve for extracting the mean transit depth, this method presented certain limitations: when concentrating on a single column of the mean-subtracted image, or a specific wavelength, it is not possible to determine the level of variation from the average value without comparative information from other channels. While incorporating adjacent channels may mitigate this limitation, we will reserve that exploration for future work. For the current investigation, we will therefore focus on the first strategy.

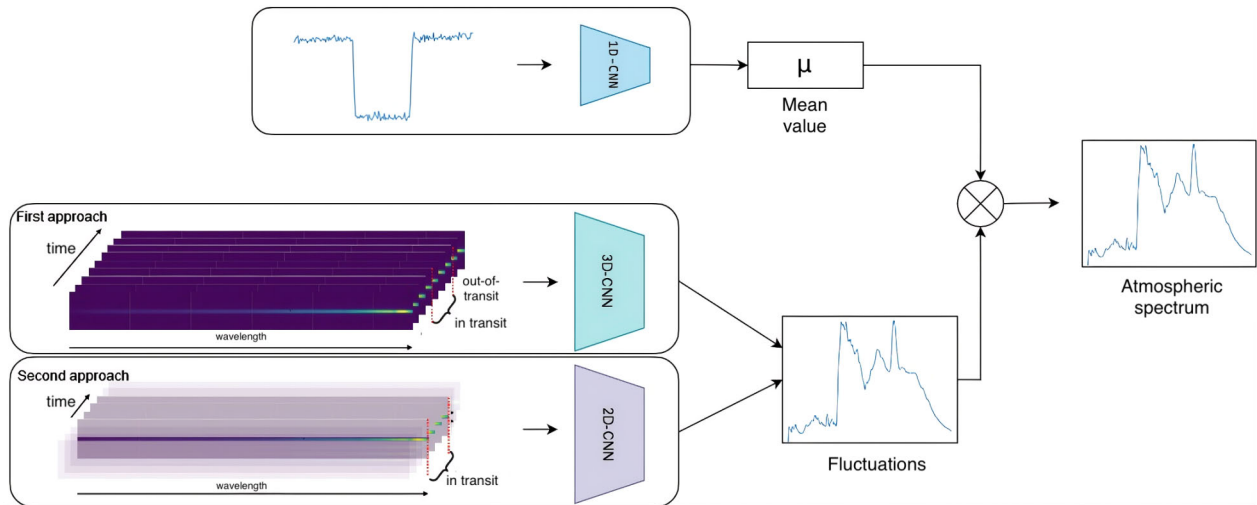
We developed two baseline approaches for the first strategy, each leveraging different features of the data set, without separating the image columns. Both baselines utilize a CNN-based architecture, but for different views of the data. A CNN is a neural network architecture that gained widespread popularity since its successful application to the ImageNet competition in 2012. The algorithm's design allows it to recognize patterns from structured data (in this case, spatial-temporal data) through hierarchical layers of learnable filters.

Both approaches employ a two-stage training process. The first stage is identical for both, utilizing a 1D-Convolutional Neural Networks (1D-CNN) to fit the mean value of the transmission spectra (see Table 6).

This CNN takes as input the transit white light curve, which represents the total flux of each image as a function of time. The second stage, however, diverges into two different approaches, each designed to capture different aspects of the data (Fig. 15.):



**Figure 14.** Stellar normalization of the detector images. Top: raw flux before normalization, showing the strongly wavelength-dependent stellar continuum concentrated along the spatial trace. Bottom: after dividing by the out-of-transit stellar spectrum, equalizing flux across wavelengths and revealing the spatial PSF profile. Left: 2D detector images (spatial pixels  $\times$  wavelength bins); right: corresponding 3D surface representations.



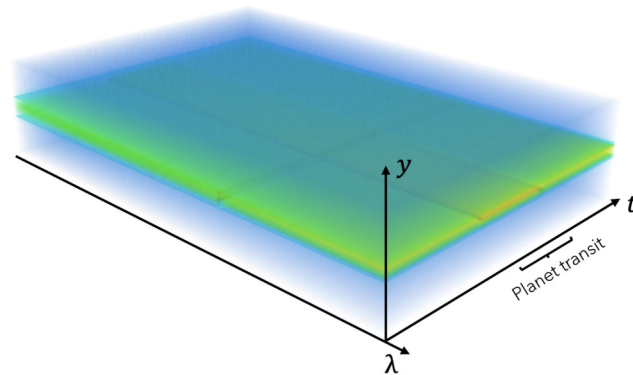
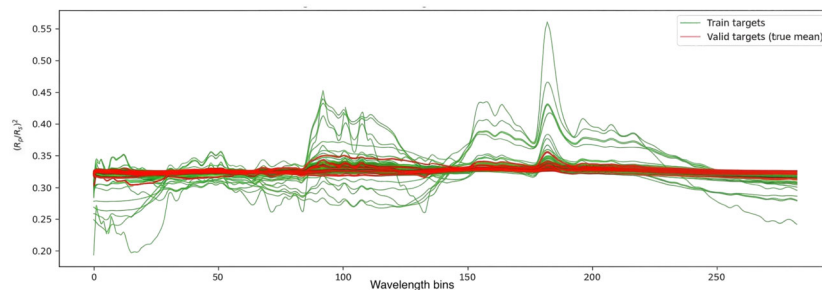
**Figure 15.** The figure illustrates the two different ML strategies deployed in this investigation. The two approaches use the same initial step (fit with a 1D CNN top row) followed by two different architectures (3D-CNN and 2D-CNN) to tackle the problem using different inputs.

(i) *First approach*: train on the 3D data cubes ( $N \times N_{\text{times}} \times N_{\text{spatial}} \times N_{\text{wavelengths}}$ ), this approach uses all the spatial information but as a consequence requires a lot of computing resources. The model is trained on a 3D CNN to search for atmospheric features.

(ii) *Second approach*: pre-process the data by summing up the fluxes along the pixel y-axis, for each wavelength, resulting in 2D images of dimension ( $N_{\text{times}}, N_{\text{wavelengths}}$ ), and transforming the images in order to enhance transit depth variations between wavelengths.

**Table 6.** Architecture of the 1D convolutional neural network (CNN) used to retrieve the mean RpRs from the white light curve. The network takes a fixed-length time-series input of 187 time-steps and outputs a single scalar estimate of the mean transit depth.

| Layer                                              | Type               | Output shape   | Kernel size |
|----------------------------------------------------|--------------------|----------------|-------------|
| <i>Feature extraction – 1D convolutional block</i> |                    |                |             |
| Input                                              | InputLayer         | $(N, 187, 1)$  | –           |
| Conv1D-1                                           | Conv1D (ReLU)      | $(N, 183, 32)$ | 5           |
| MaxPool-1                                          | MaxPooling1D       | $(N, 91, 32)$  | 2           |
| BatchNorm                                          | BatchNormalisation | $(N, 91, 32)$  | –           |
| Conv1D-2                                           | Conv1D (ReLU)      | $(N, 89, 64)$  | 3           |
| MaxPool-2                                          | MaxPooling1D       | $(N, 44, 64)$  | 2           |
| Conv1D-3                                           | Conv1D (ReLU)      | $(N, 42, 64)$  | 3           |
| MaxPool-3                                          | MaxPooling1D       | $(N, 21, 64)$  | 2           |
| Conv1D-4                                           | Conv1D (ReLU)      | $(N, 19, 128)$ | 3           |
| MaxPool-4                                          | MaxPooling1D       | $(N, 9, 128)$  | 2           |
| <i>Regression head – fully connected block</i>     |                    |                |             |
| Flatten                                            | Flatten            | $(N, 1152)$    | –           |
| Dense-1                                            | Dense (ReLU)       | $(N, 500)$     | –           |
| Dropout-1                                          | Dropout            | $(N, 500)$     | –           |
| Dense-2                                            | Dense (ReLU)       | $(N, 100)$     | –           |
| Dropout-2                                          | Dropout            | $(N, 100)$     | –           |
| Output                                             | Dense (Linear)     | $(N, 1)$       | –           |

**Figure 16.** Example of a datacube used as an input for the first baseline with the 3D-CNN. The transit depth is exaggerated for clarity and better visualization.**Figure 17.** Normalized transmission spectra  $(R_p/R_s)^2$  across wavelength bins for the 3D-CNN baseline. Green: training targets; red: validation true means.

#### 4.3 First baseline

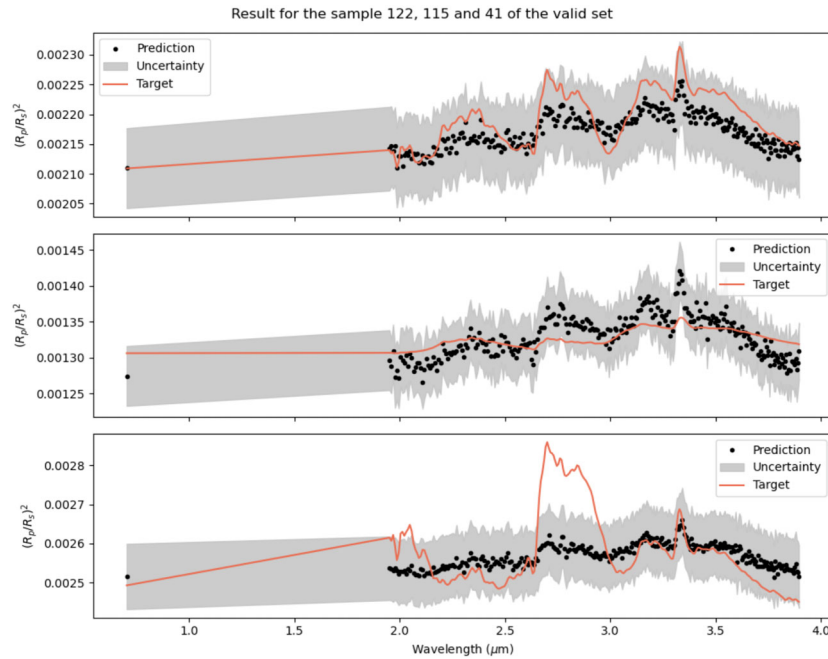
The first baseline is based on limited pre-processing of the data, which allows the neural network to use all the information contained in the spectroscopic time-series at the cost of large inputs to the model.

We outlined our treatment for the inputs, targets and architecture:

- (i) *Input*: the images are the direct product of the star spectrum normalization, as shown in Fig. 16.
- (ii) *Targets*: we remove the mean to fit atmospheric fluctuations only, and normalize the targets between 0 and 1, using the global maximum and minimum of all the examples, as shown in Fig. 17.

**Table 7.** Architecture of the first baseline. The network takes a 3D input corresponding to wavelength bins, time steps, and spectral channels, and outputs a per-wavelength fluctuation vector.

| Layer                                              | Type          | Output shape         | Kernel size           |
|----------------------------------------------------|---------------|----------------------|-----------------------|
| <i>Feature extraction – 3D convolutional block</i> |               |                      |                       |
| Input                                              | InputLayer    | (N, 283, 187, 32, 1) | –                     |
| Conv3D-1                                           | Conv3D (ReLU) | (N, 142, 94, 16, 32) | $3 \times 3 \times 3$ |
| MaxPool-1                                          | MaxPooling3D  | (N, 47, 47, 8, 32)   | $3 \times 2 \times 2$ |
| Conv3D-2                                           | Conv3D (ReLU) | (N, 24, 24, 4, 64)   | $3 \times 3 \times 3$ |
| MaxPool-2                                          | MaxPooling3D  | (N, 8, 12, 2, 64)    | $3 \times 2 \times 2$ |
| Conv3D-3                                           | Conv3D (ReLU) | (N, 8, 12, 2, 64)    | $3 \times 3 \times 3$ |
| MaxPool-3                                          | MaxPooling3D  | (N, 4, 6, 2, 64)     | $2 \times 2 \times 1$ |
| Conv3D-4                                           | Conv3D (ReLU) | (N, 4, 6, 2, 128)    | $3 \times 3 \times 3$ |
| MaxPool-4                                          | MaxPooling3D  | (N, 2, 6, 2, 128)    | $2 \times 1 \times 1$ |
| <i>Regression head – fully connected block</i>     |               |                      |                       |
| Flatten                                            | Flatten       | (N, 3072)            | –                     |
| Dense-1                                            | Dense (ReLU)  | (N, 500)             | –                     |
| Dropout                                            | Dropout       | (N, 500)             | –                     |
| Output                                             | Dense (ReLU)  | (N, 283)             | –                     |

**Figure 18.** Examples of predictions obtained using the 3D-CNN on the validation set.

(iii) *Architecture*: The 3D CNN takes in the spatial-temporal cube and outputs the atmospheric features. Detailed architecture table in Table 7.

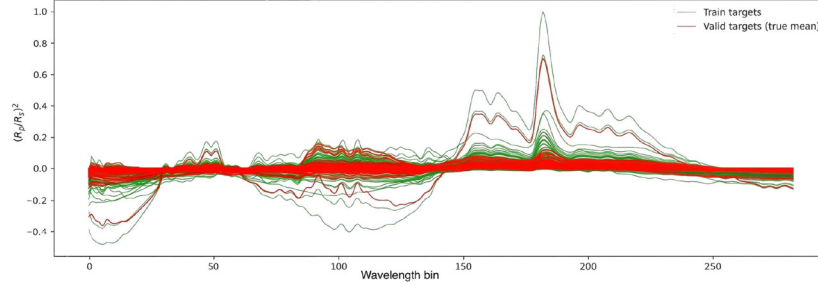
As illustrated in Fig. 18, the 3D-CNN model predicts an average spectrum, in black, that performs well across some targets but does not capture the actual atmospheric features. It adjusts only the mean value of this spectrum to match each observation but always predict the same variation pattern.

For estimating our model uncertainties and average spectra, we use Monte Carlo dropout (Y. Gal & Z. Ghahramani 2016) fixing at 1000 and 30 the number of instances for the first (1D-CNN) and second (3D-CNN) CNNs, respectively.

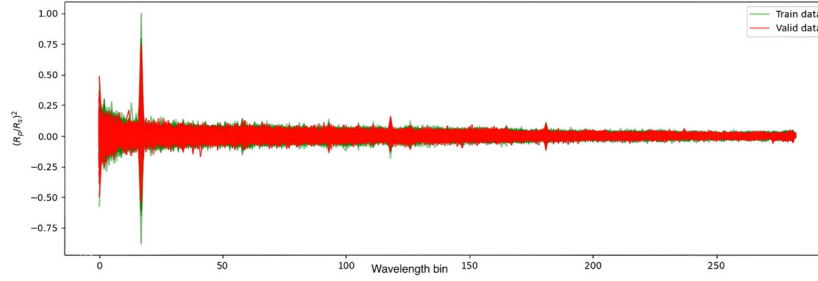
#### 4.4 Second baseline

The second baseline fits the tiny fluctuations around the mean of the spectra to search for atmospheric features. This model is also used as a baseline for NeurIPS – Ariel Data Challenge 2024 (K. H. Yip et al. 2024) and are publicly available online.<sup>11</sup> Here we propose

<sup>11</sup><https://www.kaggle.com/code/gordonyip/host-starter-solution>



**Figure 19.** Normalized transmission spectra  $(R_p/R_s)^2$  across wavelength bins for the 2D-CNN baseline. Green: training targets; red: validation true means.



**Figure 20.** Normalization of input data (2D images) for the second baseline 2D CNN, shown here for a given planet at a given time.

a method to enhance the signal variations between wavelengths, while keeping the amplitude ratios between the planetary systems, implementing the following treatment to the input:

(i) *Time range*: For the second-stage 2D-CNN input, we retain only the in-transit portion of the time-series, using the smallest transit duration in the data set to define a consistent temporal window across all examples. Note that the first-stage 1D-CNN (Section 4.2) uses the full white light curve including out-of-transit baseline to predict the mean transit depth; the 2D-CNN subsequently predicts only the spectral fluctuations around this mean value. This approach reduces the 2D-CNN input size while preserving the transit signal morphology.

(ii) *Pixel y-axis*: we sum up the fluxes along the pixel y-axis, for each wavelength, resulting in 2D images of dimension (N\_times, N\_wavelengths).

(iii) *Subtracting the mean and data normalization*: for each planetary system, we first subtract the mean signal across all wavelengths and time instants to obtain relative input data (see Fig. 19.). This pre-processing step ensures coherence with our relative target values. We then normalize these relative signals to the range  $[-1, 1]$  using the maximum amplitude of the complete data set as a reference, see Fig. 20.

(iv) *Architecture*: We implement a 2D CNN to fit the atmospheric features (see Table 8 for more details).

This approach does not use all the information contained in the datacube, e.g. the jitter effect is now accessible within the time domain only, but this implementation enables to perform fast calculations and to predict atmospheric spectra with different shapes contrary to the first baseline for which the actual implementation does not allow to produce variations in the spectral shape. For estimating our model uncertainties and average spectra, we again used the same configuration as the first approach use Monte Carlo dropout (Y. Gal & Z. Ghahramani 2016) fixing at 1000 and 30 the number of instances for the first and second CNNs respectively.

#### 4.5 Evaluation metric

We evaluate the quality of the predicted spectra,  $(\tilde{x}_\lambda)$ , and their predicted uncertainties at different wavelengths,  $(\tilde{\sigma})_\lambda$ ,  $\tilde{S} = (\tilde{x}_\lambda, \tilde{\sigma}_\lambda)$  against the ground truth spectrum,  $x_\lambda$  using two metrics, (1) Mean Squared Error (MSE) and (2) Gaussian Log-Likelihood to incorporate the estimated uncertainties from the network.

The Mean Squared Error (MSE) is calculated by taking the squared difference between the ground truth spectra and the predicted spectra for each example. We then compute the mean of these squared differences for each pair of wavelengths across all examples.

The Gaussian Log-Likelihood (GLL) based scoring function, we will calculate the Gaussian Log-Likelihood (GLL) value for each pair of  $x$  in  $S_i$  and  $\tilde{S}_i$ , i.e.

$$\text{GLL}(x_\lambda, \tilde{x}_\lambda, \tilde{\sigma}_\lambda) = -\frac{1}{2} \log(2\pi \tilde{\sigma}_\lambda^2) - \frac{(\tilde{x}_\lambda - x_\lambda)^2}{2\tilde{\sigma}_\lambda^2}. \quad (13)$$



**Table 8.** Architecture of the 2D convolutional neural network (CNN) used as a second baseline to model spectral light-curve fluctuations. The input shape corresponds to the 40 in-transit time-steps across 283 wavelength bins. The network outputs a per-wavelength fluctuation vector of length 283.

| Layer                                                           | Type               | Output shape      | Kernel size |
|-----------------------------------------------------------------|--------------------|-------------------|-------------|
| <i>Block I – Temporal feature extraction (time axis)</i>        |                    |                   |             |
| Input                                                           | InputLayer         | (N, 40, 283, 1)   | –           |
| Conv2D-1                                                        | Conv2D (ReLU)      | (N, 40, 283, 32)  | 1 × 3       |
| MaxPool-1                                                       | MaxPooling2D       | (N, 20, 283, 32)  | 2 × 1       |
| BatchNorm-1                                                     | BatchNormalisation | (N, 20, 283, 32)  | –           |
| Conv2D-2                                                        | Conv2D (ReLU)      | (N, 20, 283, 64)  | 1 × 3       |
| MaxPool-2                                                       | MaxPooling2D       | (N, 10, 283, 64)  | 2 × 1       |
| Conv2D-3                                                        | Conv2D (ReLU)      | (N, 10, 283, 128) | 1 × 3       |
| MaxPool-3                                                       | MaxPooling2D       | (N, 5, 283, 128)  | 2 × 1       |
| Conv2D-4                                                        | Conv2D (ReLU)      | (N, 5, 283, 256)  | 1 × 3       |
| Conv2D-5                                                        | Conv2D (ReLU)      | (N, 5, 283, 32)   | 1 × 3       |
| <i>Block II – Spectral feature extraction (wavelength axis)</i> |                    |                   |             |
| MaxPool-4                                                       | MaxPooling2D       | (N, 5, 141, 32)   | 1 × 2       |
| BatchNorm-2                                                     | BatchNormalisation | (N, 5, 141, 32)   | –           |
| Conv2D-6                                                        | Conv2D (ReLU)      | (N, 5, 141, 64)   | 1 × 3       |
| MaxPool-5                                                       | MaxPooling2D       | (N, 5, 70, 64)    | 1 × 2       |
| Conv2D-7                                                        | Conv2D (ReLU)      | (N, 5, 70, 128)   | 1 × 3       |
| MaxPool-6                                                       | MaxPooling2D       | (N, 5, 35, 128)   | 1 × 2       |
| Conv2D-8                                                        | Conv2D (ReLU)      | (N, 5, 35, 256)   | 1 × 3       |
| MaxPool-7                                                       | MaxPooling2D       | (N, 5, 17, 256)   | 1 × 2       |
| <i>Regression head – fully connected block</i>                  |                    |                   |             |
| Flatten                                                         | Flatten            | (N, 21760)        | –           |
| Dense-1                                                         | Dense (ReLU)       | (N, 700)          | –           |
| Dropout                                                         | Dropout            | (N, 700)          | –           |
| Output                                                          | Dense (Linear)     | (N, 283)          | –           |

**Table 9.** Results for the two baselines on validation and test sets on GLL based metric and MSE metric.

|                 | Valid GLL<br>score | Valid MSE<br>(ppm) | Test GLL<br>score | Test MSE<br>(ppm) |
|-----------------|--------------------|--------------------|-------------------|-------------------|
| 3D-CNN Baseline | 0.4681             | 128                | 0.4232            | 544               |
| 2D-CNN Baseline | 0.5020             | 86                 | 0.4246            | 542               |

The GLL values from each point will then be summed across all wavelengths and across the entire test set via  $\mathcal{L} = \sum_i \sum_\lambda \text{GLL}(x_{i,\lambda}, \tilde{x}_{i,\lambda}, \sigma_{i,\lambda})$ . A score will be generated for a given prediction set  $\tilde{\mathcal{S}}_i$  via the following formulation:

$$\text{score} = \min \left[ 0, \frac{\mathcal{L} - \mathcal{L}_{\text{naive}}}{\mathcal{L}_{\text{ideal}} - \mathcal{L}_{\text{naive}}} \right].$$

Here we denote  $\mathcal{L}_{\text{ideal}}$  as the optimum prediction set, for which  $(\tilde{x}_\lambda) = (x_\lambda)$ , and with  $\tilde{\sigma} = 10\text{ppm}$ . This uncertainty is lower than the minimum achievable theoretical uncertainty.

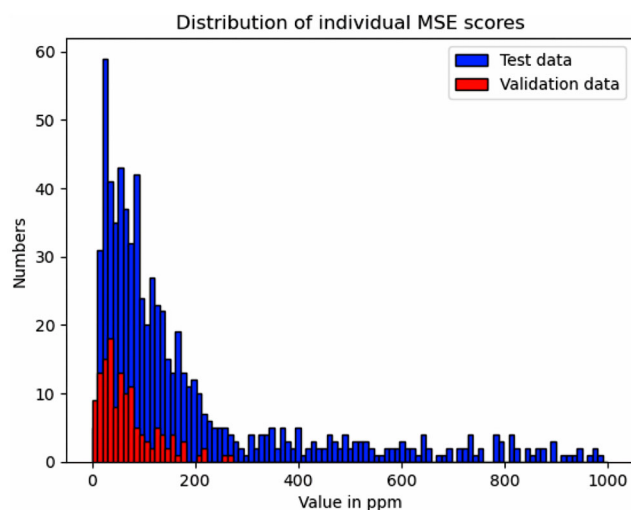
The score will be returned as a decimal in the interval [0,1], with higher scores corresponding to better performing models.  $\mathcal{L}_{\text{naive}}$  represents the reference score when the model always outputs the overall mean and standard deviation of the training set as its prediction.

## 5 RESULTS FROM ML BASELINE

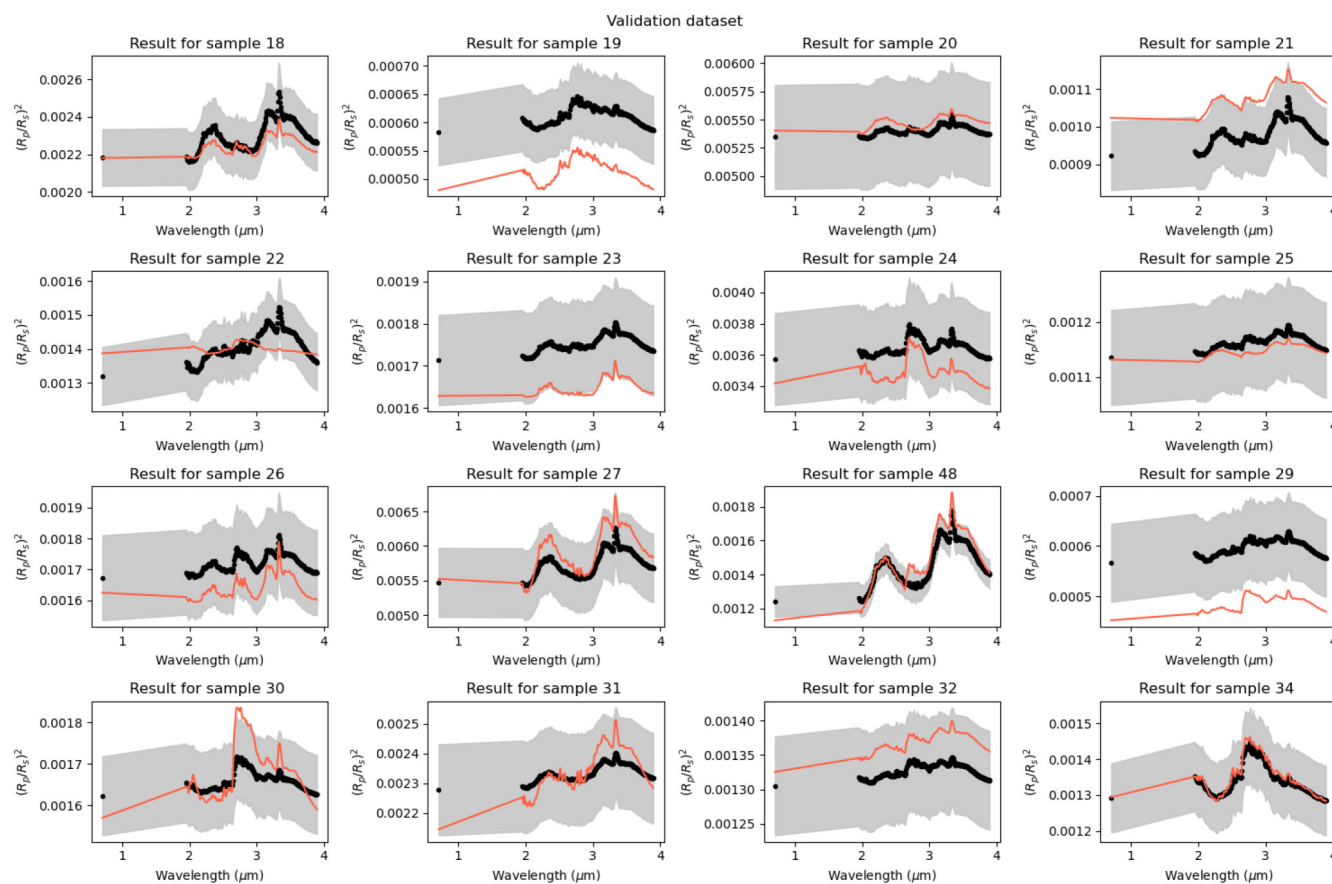
The resulting scores for the two approaches are given in Table 9 for the validation and test sets. We evaluate the models with both metric (GLL metric and MSE). For the mean computation, the SGD optimizer gives more permissive error bar on the GLL loss. Adam optimizer is used for predicting the fluctuations.

The results in Table 9 were obtained by keeping the best out of four different splits. These splits are identical across the two baselines, both sharing the first 1D CNN. We trained the models using the TRex computer from CNES, with a GPU device (V100). The computing times were 1478s and 183s for the first (3D-CNN) and second (2D-CNN) baselines respectively. Test scores are similar for the two models, whereas the validation score is usually better for the second baseline.

The first baseline model tend to converge to a single spectral shape that represents a compromise across the entire data set. In contrast, the second baseline model predicts diverse shapes for the validation set. Neither model, however, successfully generalizes to the test set. The distribution of MSE scores for the second baseline model across both validation and test populations is shown in



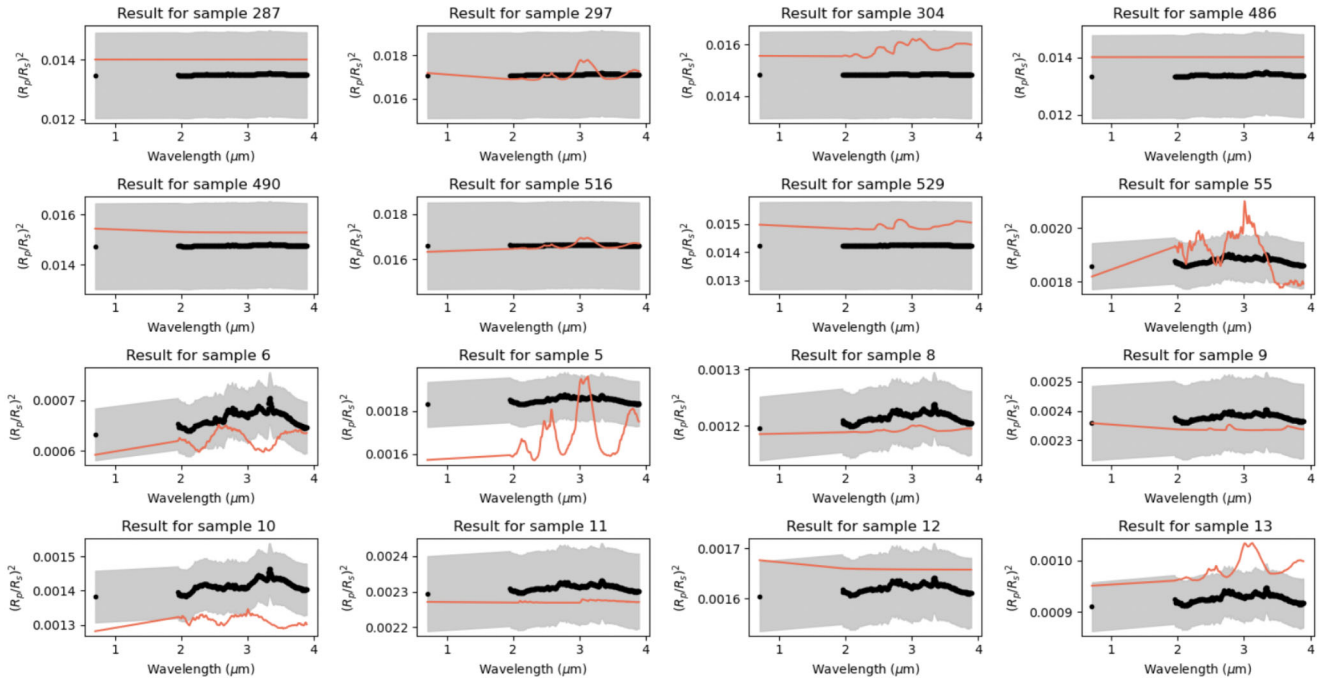
**Figure 21.** MSE distribution for validation and test data (second baseline). 35 test MSE higher than 1000 are not captured in this plot, reaching a maximum of 3440.



**Figure 22.** Model predictions from the second baseline across 16 randomly selected validation targets. Black dots/line: predicted mean spectrum; gray band: predicted uncertainty envelope; red line: true target spectrum.

Fig. 21. Random examples of predicted spectra compared to ground truth values are presented in Figs 22 and 23 for the validation and test sets, respectively.

Model performance is primarily determined by two factors. First, the ability to fit the mean spectral value significantly influences the scores. Secondly, the atmospheric extent relative to  $(R_p/R_s)^2$  affects the magnitude of spectral fluctuations, and therefore on the overall score, e.g. thinner atmospheres result in reduced modulation to the flat line. Additionally, the error bars substantially influence the GLL based metric, with larger error bars generally yielding better scores up to a threshold value.



**Figure 23.** Model predictions from the second baseline across 16 randomly selected test targets. Black dots/line: predicted mean spectrum; grey band: predicted uncertainty envelope; red line: true target spectrum.

## 6 DISCUSSION

We have presented a comprehensive public data set of simulated Ariel observations and demonstrated baseline machine learning approaches for data reduction. While our Convolutional Neural Networks successfully extract mean transit depths, they also reveal fundamental limitations when applied to test scenarios that differ systematically from training data – a challenge central to exoplanet survey science where target properties are inherently unknown.

This section examines the factors contributing to performance degradation between validation and test sets, explores mitigation strategies, and discusses the implications for developing robust data reduction pipelines for space-based transit spectroscopy missions.

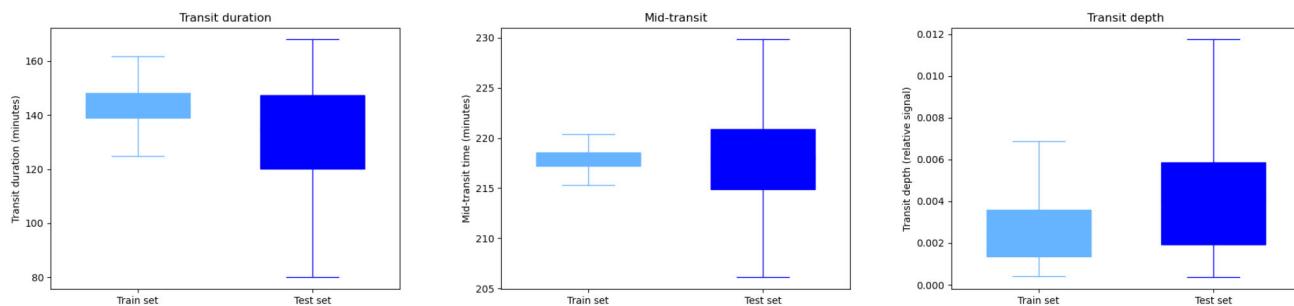
### 6.1 Performance degradation

From the result section we can see a stark performance degradation from training data (as well as validation data) to test data. This is an expected outcome stemming from the way the data set is constructed (see Table 3 for more details on each scenarios). To reiterate, we would like to explore the generalization ability of the trained algorithm in unknown scenarios. In the following paragraphs we will look into possible causes that could contribute to the degradation.

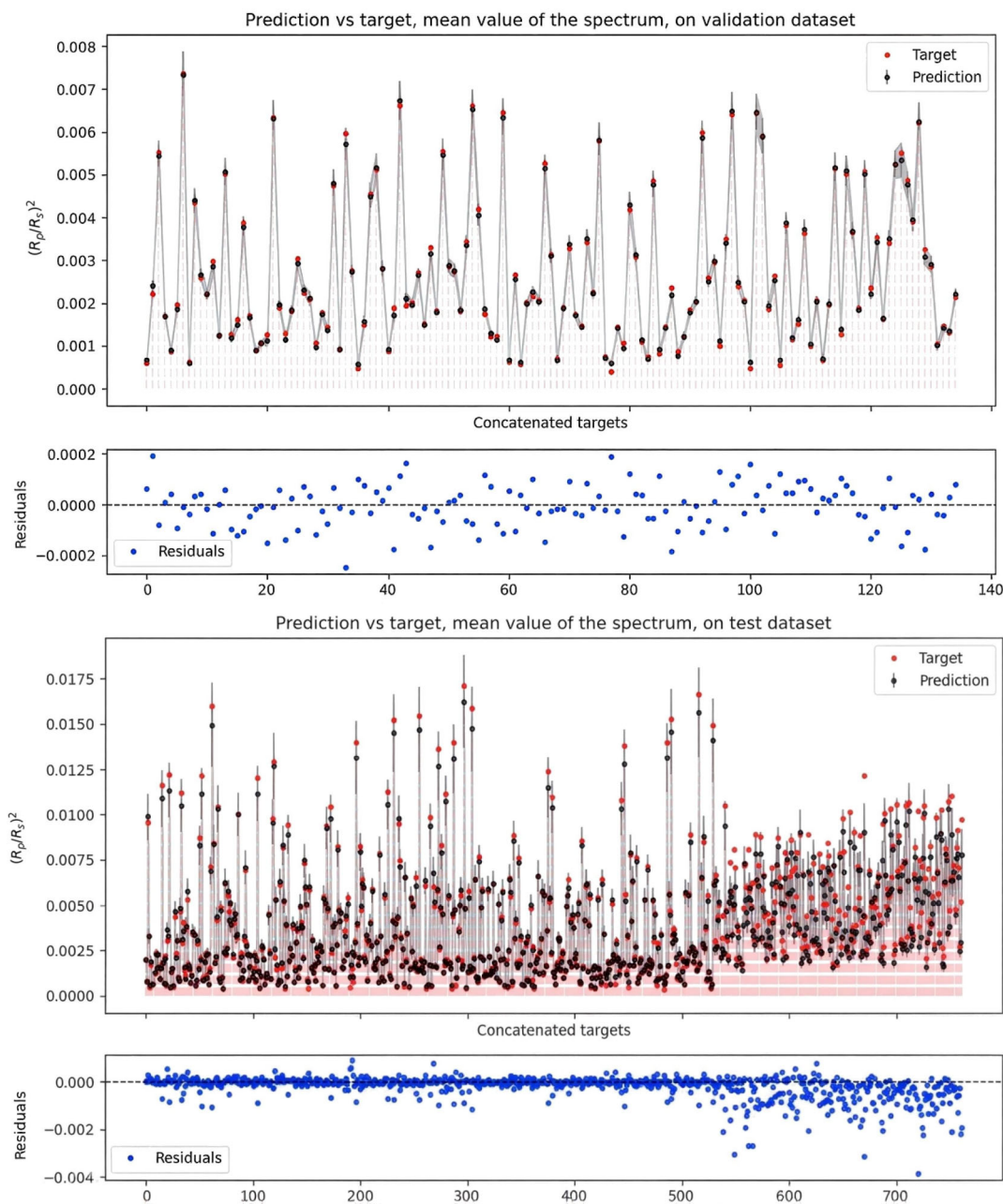
#### 6.1.1 Distribution shifts from training to test data

**Mean values:** Both the training and test data sets show variations in transit duration, mid-transit time, and transit depth within their respective population. Fig. 24 displays these parameters distribution for training (cyan) and test data (blue), revealing that even for three parameters, the distribution of the test data is broader and exhibits more extreme minimum and maximum values. Within the test data, about 35 planetary systems have extremely deep transits compared to the training population, resulting in a 5 – 10 per cent error or more in our predicted mean values, as shown in Fig. 25. Likewise, the predictions are degraded when the transits are narrower or shifted in time compared to the training set range. Given this shift in transit and atmospheric characteristics between training and test sets, uncertainties of the mean values for the test set are often underestimated while they could be overestimated for the validation set.

**Atmospheric features:** As presented in Table 3, the atmospheric composition of planets that belong to the test data is different from the training data. This results in different atmospheric features in their spectra. Fig. 26 shows the normalized fluctuations of the test set spectra that can be compared to the ones exhibited in Fig. 20 of the train and validation sets. The significant shift between the atmospheric features in the training and test data sets poses a significant challenge to the model's generalization capabilities, as it encounters a markedly different distribution during evaluation. Such distribution is never presented in the training set means that the model cannot simply perform well by learning to memorize the pattern and recognize it during evaluation. Instead, it is forced to

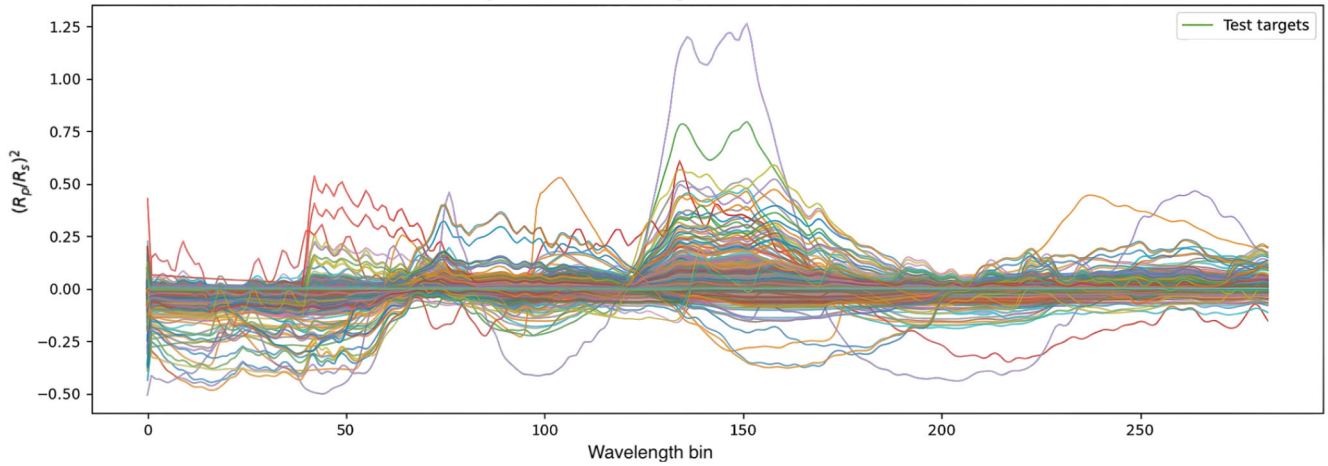


**Figure 24.** Distribution of transit duration, mid-transit time, and transit depth, for train and test sets, represented by shades of blue. The solid rectangle is centred on the mean value, and its height is defined by the standard deviation. The whiskers extend to the minimum and maximum values, representing the range of the data. Time indices are converted into minutes (see conversion in Section 4.1).



**Figure 25.** Predictions of spectra mean values for validation (top) and test (bottom) data, with error bars (black dots). Ground Truth mean value are represented by red dots and residuals are shown in the lower part of each figure (blue points).





**Figure 26.** Test set transmission spectra after subtracting from the population median, highlighting relative atmospheric features across the wavelength range.

perform inference based on the incoming data alone. This distribution shift acts as a fundamental constraint on the model’s ability to perform well on unseen data.

### 6.1.2 Small training set

The data set design aligns with conservative estimates of Ariel’s science yield, but its size (1453 samples) presents a significant challenge for deep learning applications. Deep learning models derive their power from approximating arbitrary functions through substantial numbers of tunable parameters. Given the dense information content and noise within each sample, the data dimensionality necessitates a large parameter space. For example, our CNN architectures require 15M parameters – orders of magnitude larger than the number of training examples. This parameter-to-sample ratio impedes network convergence to generalizable solutions and increases susceptibility to overfitting.

### 6.1.3 Uncertainty quantification

Reliable uncertainty quantification in neural networks remains a significant challenge in deep learning research. The high dimensionality of neural network parameter spaces complicates error propagation tracking, yet reliable confidence estimates are crucial for scientific applications. Current solutions typically rely on post-hoc approaches, such as the MC Dropout method implemented in this study.

Our results reveal two fundamental limitations of this approach. First, comparing Figs 22 and 23 shows minimal variation in error bar magnitude between validation and test sets, suggesting the model learned to replicate the training set’s error distribution without adapting to the test distribution. Secondly, Fig. 25 reveals that predicted uncertainties often exceed actual residuals for many validation examples while simultaneously failing to scale appropriately for out-of-distribution test examples with large prediction errors. In both regimes – whether predictions are accurate or inaccurate – the uncertainties remain similarly sized, indicating the method cannot distinguish between familiar and unfamiliar inputs.

This asymmetric behaviour indicates that Monte Carlo (MC) Dropout learns the uncertainty distribution characteristic of the training data but cannot reliably extrapolate to novel scenarios. The method effectively memorizes the typical scatter in the training distribution rather than providing true epistemic uncertainty estimates that would appropriately increase for unfamiliar inputs. Ideally, the model should produce larger error bars for unseen examples, reflecting increased uncertainty. Instead, it applies learned uncertainty patterns to novel cases, adversely affecting the Gaussian Log Likelihood based score (GLL-score). This limitation is particularly problematic for survey missions like Ariel, where observations will inevitably encounter planetary systems outside the parameter space of any training set.

## 6.2 Mitigation

In this section we will discuss possible mitigations to the observed degradation in the previous section.

### 6.2.1 Ad-hoc scaling factor

Performance degradation in the test set can be attributed to underestimated error bars, resulting in severe scoring penalties. Applying differential scaling factors ( $f = 0.7$  for validation and  $f = 1.4$  for test sets) improved the second baseline scores to 0.5192 and 0.4570, re-



spectively. While this adjustment improves model alignment with the test set, such data set-specific modifications are not recommended as it generally does not scale well beyond the provided data set. Notably, the model underestimates uncertainties even within its training distribution, indicating a fundamental limitation in uncertainty estimation methods that warrants future research.

### 6.2.2 Choice of optimizers

Our investigation revealed a significant performance difference between optimization algorithms. The Adam optimizer, despite its popularity for fast convergence and training stability, failed to achieve positive GLL-based metrics on the test set. Switching to SGD<sup>12</sup> substantially improved generalization, achieving a test score of  $\sim 0.45$ . This behaviour aligns with existing literature (M. Hardt, B. Recht & Y. Singer 2016; A. C. Wilson et al. 2017), supporting evidence that SGD's stochastic nature provides better generalization performance on unseen data despite slower convergence.

## 6.3 Supplementing test data to the training set

We investigated the impact of incorporating limited test data into the training set, simulating early mission phases where initial observations become available. Table 10 presents a quantitative analysis of performance across different training-test mixtures, progressively adding random test samples (50, 100, 150, 400, and 700 out of 762 total test examples) to the original 576-sample training set.

Performance improvements show two distinct regimes. The MSE metric exhibits rapid initial improvement: adding just 50 test samples (8.6 per cent of available test data) reduces test MSE from 542 to 271 ppm, nearly a factor of two. This dramatic improvement suggests that even limited exposure to the test distribution helps the model learn critical features of the shifted domain. However, further additions show diminishing returns: increasing from 50 to 400 samples provides only incremental MSE reduction (271 to 185 ppm), and the curve appears to plateau unless the majority of test data is included.

GLL scores show a different pattern. While MSE improvements are front-loaded (large gains from first 50 samples), GLL improvements are more gradual and persist across the full range of added samples. Even with 700 test samples added, the GLL score (0.4945) remains below the validation set performance (0.5020 in the original split). This divergence indicates that *accuracy* (MSE) improves faster than *calibration* (GLL).

Implications for uncertainty quantification: The MSE-GLL divergence reveals that the network learns to predict spectral values more accurately but fails to appropriately adjust its uncertainty estimates. As prediction errors decrease with added test data, the model's error bars should correspondingly narrow, but they remain approximately constant, leading to overconfident predictions. This results in good MSE performance but continued GLL penalties, reinforcing our findings in Section 6.1.3 that Monte Carlo Dropout provides poorly calibrated uncertainties under distribution shift.

These results suggest that adaptive learning strategies incorporating even small amounts of mission data could substantially improve pipeline performance, but improved uncertainty quantification methods will be necessary to achieve well-calibrated confidence estimates.

## 6.4 Limitations

While the data set and methodology presented in this work represent a significant step forward in the preparation for Ariel's data reduction challenges, a number of limitations must be acknowledged to frame the scope and applicability of the results.

First, the simulated data set, although based on updated payload configurations and realistic noise sources, including those derived from *JWST* calibration products, inevitably relies on a set of assumptions and simplifications. For instance, the pointing jitter is derived from a single realization of a timeline, which is randomly shifted across simulations but may not fully encapsulate the range and variability expected during the mission's operational lifetime. This constraint could limit the generality of conclusions drawn from jitter-related analyses.

Moreover, the atmospheric models used to generate the transit spectra adopt a simplified, isothermal, 1D structure with constant chemical abundances and a fixed set of trace gases. These models are intentionally not physically self-consistent in order to challenge the robustness of data reduction pipelines under diverse scenarios. However, this choice also means that certain effects, such as vertical chemical gradients, disequilibrium chemistry, or cloud heterogeneity, are not represented, reducing the astrophysical realism of the simulations.

Another important limitation concerns the stellar sample used in this exercise. The data set includes only four host stars, selected for their brightness and compatibility with the imposed observing strategy. While this choice ensures consistency and allows for short integration times suitable for jitter analysis, it significantly restricts the diversity of stellar types and SEDs present in the data set. Moreover, the sample omits both limb-darkening effects and the presence of stellar spots-phenomena that significantly alter transit

<sup>12</sup>Stochastic Gradient Descent (SGD) is an iterative optimization algorithm that updates neural network parameters by computing gradients on small random subsets (mini-batches) of the training data, rather than the entire data set. This introduces noise into the optimization process, which can help escape local minima and improve generalization.

**Table 10.** Scores when adding some test data in the training set.

|                           | Original scores |      | With 50 test data |      | With 100 test data |      | With 150 test data |      | With 400 test data |      | With 700 test data |      |
|---------------------------|-----------------|------|-------------------|------|--------------------|------|--------------------|------|--------------------|------|--------------------|------|
|                           | Valid           | Test | Valid             | Test | Valid              | Test | Valid              | Test | Valid              | Test | Valid              | Test |
| MSE mean values (in ppm)  | 79              | 530  | 97                | 254  | 137                | 259  | 117                | 171  | 145                | 175  | 127                | 140  |
| MSE fluctuations (in ppm) | 40              | 110  | 49                | 95   | 49                 | 86   | 43                 | 67   | 45                 | 61   | 50                 | 55   |
| MSE total (in ppm)        | 88              | 542  | 108               | 271  | 145                | 273  | 125                | 184  | 151                | 185  | 136                | 150  |
| GLL score                 | 5016            | 4246 | 4560              | 4912 | 4438               | 4702 | 4346               | 4860 | 4592               | 4880 | 4674               | 4945 |

morphology. As a result, the generalizability of machine learning models trained on this data set to a broader population of stars remains untested and should be approached with caution.

The observational setup is also standardized across the entire data set, with fixed integration times, transit durations, and observing windows. While this uniformity supports the implementation and training of machine learning pipelines, it sacrifices the variability inherent in actual observations, where exposure times and transit coverage are tailored to individual targets. Consequently, the data set may not fully capture the complexity of real-time scheduling and its implications for noise and signal extraction.

In terms of calibration, the decision to perturb the calibration products uniformly by 5 per cent simulates uncertainties in a controlled fashion, but does not reproduce the full complexity and temporal evolution of real calibration errors, such as detector degradation, flat-field instability, or long-term non-linear effects.

Finally, while the baseline neural network models presented in this study offer a proof-of-concept for learning-based reduction strategies, they are not optimized for scientific use. The training data set remains limited in terms of astrophysical diversity and instrumental conditions, and the neural network architectures do not exploit advanced strategies such as self-supervised learning, transfer learning, or domain adaptation. Moreover, the risk associated with domain shift, that is, the discrepancy between the training and deployment data distributions, remains an open challenge and is only partially addressed through the proposed data partitioning strategy.

Despite these limitations, the data set provides a valuable and well-characterized platform for benchmarking data reduction pipelines. It has been designed with modularity and extensibility in mind, enabling future developments that may address the limitations outlined above.

## 7 CONCLUSION

This work presents a comprehensive public data set of simulated Ariel mission observations, addressing a critical need in the exoplanetary atmospheric characterization community. By combining ExoSim2 and TauREx3 to create realistic synthetic data, we have developed one of the most extensive benchmark data sets based on the current payload design of the ESA Ariel Space Mission. The data set includes 1435 simulated observations across diverse stellar and planetary configurations, featuring realistic noise models, calibration products from *JWST* NIRSpec observations, and systematic effects based on current space-based missions.

We provide, as an initial attempt, the use of deep learning models for a potentially scalable data reduction pipeline. We implemented and evaluated two baseline deep learning approaches demonstrating both the potential and challenges of using an end-to-end ML-based data reduction pipelines.

While these models proved to be quite successful in extracting the average transit depths from noisy data, they also highlight critical issues with non-parametric models such as the tendency to overfit on the training data sets, unable to mitigate distribution shifts between training and test data and unreliable uncertainty quantification. These findings highlight the importance of developing robust, adaptable algorithms capable of handling diverse observational scenarios.

We have also investigated the impact of data set composition on the ability for the model to generalize. Our analysis reveals that even small additions of test data to the training set can significantly improve model performance, hinting that adaptive learning strategies, or continuous learning strategies may be useful to optimize the reliability of these algorithms. The data set has already proven its value through the Ariel Data Challenge 2024, providing a standardized and open platform for algorithm development and comparison.

Moving forward, this resource will support the broader scientific community in developing scalable, efficient, and reliable data reduction techniques essential for maximizing the scientific return of the Ariel mission and future exoplanet characterization efforts. As we approach the launch of the *Ariel Space Telescope* in the coming future, continued refinement of these methods will be crucial for achieving the mission's ambitious goals in comparative planetology and atmospheric science.

### 7.1 Going beyond the data challenge

Although the data set was originally released in the context of the Ariel Data Challenge 2024, its structure, complexity, and realism render it suitable for a wide range of applications beyond the scope of the competition. Designed to replicate the full observational process, including time, wavelength, and spatially dependent noise sources and calibration uncertainties, the data set can serve as a valuable benchmark for the community.

First, the data set can be used as a benchmark for the development and testing of data reduction pipelines. Its design incorporates a broad range of realistic noise sources, such as gain drifts, pointing jitter, pixel non-linearity, and detector-level systematics, making it a valuable resource for evaluating the performance and robustness of classical, Bayesian, and machine learning-based approaches under conditions representative of actual space-based observations.

A key feature of the data set is the inclusion of the ground truth atmospheric spectra. This enables rigorous quantification of systematic biases introduced during the data reduction process. By comparing extracted transmission spectra to the known input values, users can evaluate the absolute accuracy of their methods and identify wavelength-dependent residuals or distortions, providing critical feedback for pipeline optimization.

The data set also provides an ideal testbed for methodological studies in machine learning, such as domain generalization, transfer learning, and anomaly detection. The structured shift between training and test distributions has been explicitly designed to reflect realistic deployment scenarios, offering an opportunity to assess model generalizability in the presence of non-stationary noise. Ultimately, the extraction of planetary spectra from photometric and spectroscopic light curves for each Ariel target will balance target-specific adjustments with a uniformly applied reduction framework across the entire sample. In this context, machine-learning methods will play a key role in automatically detecting anomalous cases – such as those impacted by stellar contamination, thus flagging instances that may require more bespoke treatment.

Finally, given its accessibility and comprehensive documentation, the public data set is well-suited for training and educational purposes. It can be employed to demonstrate the complexities of exoplanet transit spectroscopy, the role of calibration products, and the importance of error propagation in atmospheric retrievals.

Overall, the data set aims to support the development of scalable, accurate, and reproducible methodologies for exoplanetary atmospheric characterization, and to foster community readiness for the analysis of Ariel data and other forthcoming space-based surveys.

## ACKNOWLEDGEMENTS

The work presented in this paper was partially supported by UKSA, grant ST/X002616/1, ST/W002507/1, and by the Italian Space Agency (ASI) with the Ariel grant n. 2021.5.HH.0. This work used the DiRAC Data Intensive service (CSD3) at the University of Cambridge, managed by the University of Cambridge University Information Services on behalf of the STFC DiRAC HPC Facility ([www.dirac.ac.uk](http://www.dirac.ac.uk)). The DiRAC component of CSD3 at Cambridge was funded by BEIS, UKRI, and STFC capital funding and STFC operations grants. DiRAC is part of the UKRI Digital Research Infrastructure.

The authors also wish to thank the Ariel S2MD Working Group for the technical support and input on the noise simulations.

## CONFLICTS OF INTEREST

Authors declare no conflict of interest.

## DATA AVAILABILITY

The train data generated for this paper are available at <https://www.kaggle.com/competitions/ariel-data-challenge-2024/data>.

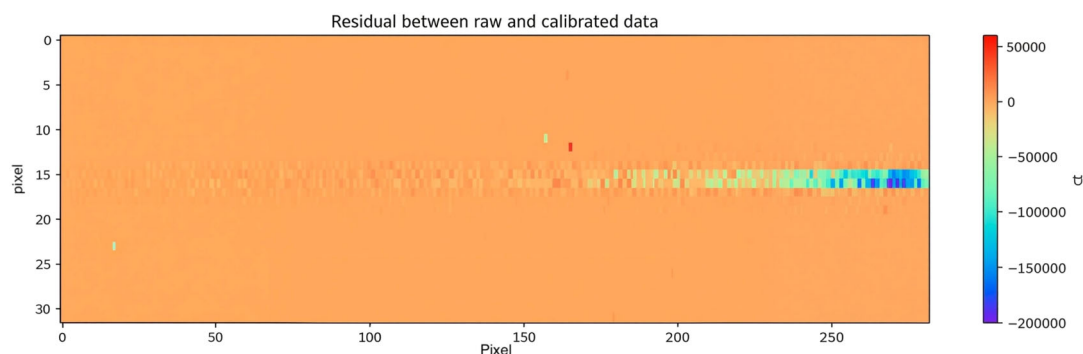
## REFERENCES

- Ahrer E.-M. et al., 2023, *Nature*, 614, 653  
 Al-Refaie A. F., Changeat Q., Waldmann I. P., Tinetti G., 2021, *ApJ*, 917, 37  
 Alderson L. et al., 2023, *Nature*, 614, 664  
 Azzam A. A. A., Tennyson J., Yurchenko S. N., Naumenko O. V., 2016, *MNRAS*, 460, 4063  
 Baraffe I., Homeier D., Allard F., Chabrier G., 2015, *A&A*, 577, A42  
 Barber R. J. et al., 2014, *MNRAS*, 437, 1828  
 Barron N. et al., 2007, *PASP*, 119, 466  
 Bie F. et al., 2025, *IEEE Trans. Artif. Intell.*, 47, 3  
 Bocchieri A., Mugnai L. V., Pascale E., 2024, in Coyle L. E., Matsuura S., Perrin M. D., eds, *Proc. SPIE Conf. Ser. Vol. 13092, Space Telescopes and Instrumentation 2024: Optical, Infrared, and Millimeter Wave*. SPIE, Bellingham, p. 130924K  
 Bocchieri A. et al., 2025, *Exp. Astron.*, 59, 31  
 Buldyreva J., Yurchenko S. N., Tennyson J., 2022, *RAS Tech. Instrum.*, 1, 43  
 Carter A. L. et al., 2024, *Nat. Astron.*, 8, 1008  
 Challenger R. C. et al., 2025, *Nat. Astron.*, 9, 1821  
 Changeat Q., Yip K. H., 2023, *RAS Tech. Instrum.*, 2, 45  
 Changeat Q., Ito Y., Al-Refaie A. F., Yip K. H., Lueftinger T., 2024, *AJ*, 167, 195  
 Chubb K. L., Tennyson J., Yurchenko S. N., 2020, *MNRAS*, 493, 1531  
 Coles P. A., Yurchenko S. N., Tennyson J., 2019, *MNRAS*, 490, 4638  
 Dubey S. R., Singh S. K., 2024, *IEEE Trans. Artif. Intell.*, 5, 4851  
 Edwards B., Mugnai L., Tinetti G., Pascale E., Sarkar S., 2019, *AJ*, 156, 242  
 Elad M., Kowar B., Vaksman G., 2023, *SIAM J. Imaging Sci.*, 16, 1594  
 Espinoza N. et al., 2025, *ApJ*, 990, L52  
 Feinstein A. D. et al., 2023, *Nature*, 614, 670  
 Gal Y., Ghahramani Z., 2016, in *Proc. 33rd International Conference on Machine Learning*, Vol. 48. MLResearchPress, p. 1050  
 Greene T. P., Line M. R., Montero C., Fortney J. J., Lustig-Yaeger J., Luther K., 2016, *ApJ*, 817, 17  
 Hardt M., Recht B., Singer Y., 2016, in *International conference on machine learning*, PMLR, p. 1225  
 Holmberg M., Madhusudhan N., 2023, *MNRAS*, 524, 377  
 Ingalls J. G. et al., 2016, *AJ*, 152, 44  
 Krick J. E., Fraine J., Ingalls J., Deger S., 2020, *AJ*, 160, 99  
 Li G. et al., 2015, *ApJS*, 216, 15

- Morvan M., Nikolaou N., Tsiaras A., Waldmann I. P., 2020, *AJ*, 159, 109
- Morvan M., Nikolaou N., Yip K., Waldmann I., 2022, in Proc. 39th International Conference on Machine Learning (ICML 2022), Machine Learning for Astrophysics. p. 11
- Mugnai L. V., Pascale E., Edwards B., Papageorgiou A., Sarkar S., 2020, *Exp. Astron.*, 50, 303
- Mugnai L. V., Bocchieri A., Pascale E., Lorenzani A., Papageorgiou A., 2025, *Exp. Astron.*, 59, 9
- Nikolaou N. et al., 2023, *RAS Tech. Instrum.*, 2, 695
- Park H., Jo Y., Kang S., Kim T., Jee M. J., 2024, *ApJ*, 972, 45
- Polyansky O. L., Kyuberis A. A., Zobov N. F., Tennyson J., Yurchenko S. N., Lodi L., 2018, *MNRAS*, 480, 2597
- Radica M. et al., 2023, *MNRAS*, 524, 835
- Rothman L. S. et al., 2010, *J. Quant. Spec. Radiat. Transfer*, 111, 2139
- Rustamkulov Z. et al., 2023, *Nature*, 614, 659
- Sarkar S., Madhusudhan N., Constantinou S., Holmberg M., 2024, *MNRAS*, 531, 2731
- Schmidt S. P. et al., 2025, *AJ*, 170, 298
- Sing D. K., 2018, preprint ([arXiv:1804.07357](https://arxiv.org/abs/1804.07357))
- Stassun K. G., Collins K. A., Gaudi B. S., 2017, *AJ*, 153, 136
- Su J., Xu B., Yin H., 2022, *Neurocomputing*, 487, 46
- Sweere S. F. et al., 2022, *MNRAS*, 517, 4054
- Taylor J. et al., 2025, *MNRAS*, 540, 3677
- Thurairethinam V., Bocchieri A., Savini G., Mugnai L. V., Pascale E., 2025, *J. Astron. Tel. Instr. Syst.*, 11, 014003
- Tinetti G., Encrenaz T., Coustenis A., 2013, *A&AR*, 21, 63
- Tinetti G. et al., 2018, *Exp. Astron.*, 46, 135
- Underwood D. S., Tennyson J., Yurchenko S. N., Huang X., Schwenke D. W., Lee T. J., Clausen S., Fateev A., 2016, *MNRAS*, 459, 3890
- Vojtekova A., Lieu M., Valtchanov I., Altieri B., Old L., Chen Q., Hroch F., 2021, *MNRAS*, 503, 3204
- Welbanks L. et al., 2025, *Nat. Astron.*, 10, 234.
- Wilson A. C., Roelofs R., Stern M., Srebro N., Recht B., 2017, *Adv. Neur. Inf. Proc. Syst.*, 30, 4151
- Yip K. H., Tsiaras A., Waldmann I. P., Tinetti G., 2020, *AJ*, 160, 171
- Yip K. H. et al., 2022, preprint ([arXiv:2206.14642](https://arxiv.org/abs/2206.14642))
- Yip K. H. et al., 2024, in NeurIPS 2024 Competition Track. Available at: <https://openreview.net/forum?id=1mGm9tHrFT>
- Yurchenko S. N., Amundsen D. S., Tennyson J., Waldmann I. P., 2017, *A&A*, 605, A95
- Yurchenko S. N. et al., 2024, *MNRAS*, 527, 9736
- Zhang C., Zhang C., Zhang M., Kweon I. S., 2023, preprint ([arXiv:2303.07909](https://arxiv.org/abs/2303.07909))

## APPENDIX A: DATA CALIBRATION

Fig. A1 presents the difference between the original raw frame and the same image after calibration, including flat-field, dark current, dead pixel, and pixel non-linearity corrections. Outside the signal region, the residuals are minimal and remain close to zero, as indicated by the orange colour scale. In contrast, larger deviations are observed in regions of higher signal. This behaviour indicates that the dominant contribution to the calibration difference arises from the non-linearity correction, which becomes increasingly significant at higher count levels.



**Figure A1.** Difference between a raw image from the train set and the associated calibrated frame, applying flat-field, dark and dead pixels' correction and non-linearity of the pixels' response correction.

This paper has been typeset from a  $\text{\LaTeX}$  file prepared by the author.