2022 Ariel Data Challenge Solution: Regular Track

Team Name: The Gators

November 9 2022

Below we answer the specific questions from the organizers.

1 How did you use the training data?

- We formatted the provided spectral information, auxiliary parameters, FM-parameters, and retrieved quantile data for each planet as several Numpy data-files, which we have included in our submission, since our code reads them as inputs. If you prefer to read the original files, please uncomment the corresponding block of code, and comment out the code which imports the Numpy arrays. The trace data was used in the format given in the challenge without any changes.

- We only trained on the 21,988 labeled samples. We did not use the remaining unlabelled samples in the analysis.

- Since the objective was to produce a distribution with a small Earth Movers Distance (EMD) relative to the provided retrieved distribution, it would be natural to use the EMD in the loss function. However, this is very computationally expensive. Instead, we approximately parameterize the distribution as described below (see Section 4), and our model then predicts the parameters of this distribution. The target values for those parameters (used for the actual training) are generated by fitting to the given trace data. Note: We are including this target array with our submission, along with code to regenerate it. In order to retrain the model, you can either use the array provided, or run the code to generate a new one which is nearly identical.

2 Did you perform any data preprocessing step?

1. For planets with spectral values higher than 0.1, we replaced these anomalously high values with a value constructed from the other (lower than 0.1) spectral values.

2. In order to focus on the effect of the atmosphere, we subtract the contribution of the planet itself:

$$M'_\lambda = M_\lambda - \frac{R^2_{\text{planet}}}{R^2_{\text{star}}} \tag{1}$$

3. We used the analytical solution of the thermal equilibrium between the star and planet to estimate the temperature T_p of the planet from the auxiliary parameters alone.

4. We constructed the additional features $\frac{R_P}{R_S}$, $\frac{D}{H}$, and $\frac{R_S}{H}\max_\lambda(M'_\lambda)$, and concatenated them with the auxiliary parameters.

5. We standardized the auxiliary features and normalized M'_λ and the noise data by dividing by their respective maximum values for each planet.

3 What kind of model did you go for?

We use several fully connected neural networks some of which use concatenations or products of the outputs of previous modules as inputs.

4 What is the input/output of the model?

The model takes in the 52 wavelength bins of the flux modulation data, the 52 corresponding uncertainties (noise), the nine provided auxiliary parameters plus the three additional features we constructed. The model takes these three vectors as input, and has a total of 116 input neurons.

We parameterized the estimate of the posterior distribution in terms of 20 parameters $\{\mu_i, \sigma_i, A_i, m_i\}$, $i = 1, 2, ..., 5$, as

$$\rho(T, \vec{x}; T_p, \mu_i, \sigma_i, A_i, m_i) = \frac{1}{2}(\Theta(T_p - T)\,\mathcal{N}(T; T_p, \sigma_{T1}) + \Theta(T - T_p)\,\mathcal{N}(T; T_p, \sigma_{T2})) \times \prod_{i=1}^{5}\left[\frac{A_i}{\sigma_i\sqrt{2\pi}} e^{-\frac{(x_i-\mu_i)^2}{2\sigma_i^2}} + (1 - A_i)\frac{\Theta(x_i - 12)\Theta(m_i - x_i)}{m_i + 12}\right]$$

Here $\mathcal{N}(x; \mu, \sigma)$ is the normal distribution with mean μ and standard deviation σ as a function of x, and $\Theta(x)$ is the Heaviside step function. σ_{T1} and σ_{T2} are fitted functions proportional to T_p. The second term in the square brackets (the uniform distribution) is added to reproduce the observed effect of the prior — that when a concentration is too small to be detected, the posterior distribution ends up being uniform across all compatible concentrations.

Out model outputs these twenty numbers: $\{\mu_i, \sigma_i, A_i, m_i\}$ which are fed into the distribution along with the temperature T_p.

5 Did you do any post-processing to the output?

There is no post processing.

6 Did you perform any sampling step? If so please describe.

Given our parameterization of the distribution, we have an analytical function for the cumulative distribution function (CDF). We do inverse transform sampling using this CDF independently for each of the five concentrations. Temperature is separately assumed to be nearly Gaussian.

7 Did you use any external library and/or forward model?

We did not use any forward model. We used standard libraries such as pytorch, scipy, pandas, and numpy.