

ANKLES: Binary Orbital Parameters from Simulation Based Inference

ANYA PHILLIPS  ¹

¹ Center for Astrophysics | Harvard and Smithsonian

ABSTRACT

The upcoming Via survey will use high-precision radial velocity observations of stellar streams to search for dark matter on small scales over a duration of 5 years. In acquiring time-series radial velocity measurements, Via will provide the first insight into the binary population in globular cluster stellar streams. To recover the Keplerian orbital parameters of single-lined spectroscopic binaries as well as the parameter uncertainties, typical approaches to fitting the radial velocity curves involve Markov-Chain Monte Carlo or similar stochastic sampling techniques, which can be time-consuming for large sample sizes. In this work, we present a simulation-based inference model to recover the full posteriors on orbital parameters from the radial velocity curves. We train our model on densely-sampled RV curves as a best-case scenario, but discuss the applicability of this framework to sparsely-sampled RV curves with multi-modal posteriors to enable population-level statistics to characterize binaries observed with Via. We demonstrate the model’s successes in broadly characterizing parameter estimate uncertainties compared to a baseline multilayer perceptron. Finally, we discuss further improvements to the model so that, at the completion of the Via survey, it will enable the first statistical analysis of binary demographics in globular cluster streams.

1. INTRODUCTION

A current focus in constraining the nature of dark matter is its detection on scales $< 10^7 M_{\odot}$. These dark matter “subhalos” are predicted to exist in large numbers by standard cold dark matter cosmological models (e.g., A. Klypin et al. 1999; B. Moore et al. 1999; J. Diemand et al. 2007; V. Springel et al. 2008), but they are not expected to host luminous baryons and so they remain as yet undetected. One approach to detecting dark matter on small scales is through the kinematic signatures they imprint on the dynamically cold tidal tails of Milky Way globular clusters (“stellar streams”) during close stream-subhalo encounters (which are forecasted to be frequent; J. H. Yoon et al. 2011). From kinematic and positional observations of stellar streams, it is possible to deduce the properties of past stream/subhalo interactions such as impact parameter, relative velocity, and most importantly, the subhalo mass and size (e.g., D. Erkal & V. Belokurov 2015). Inferring the masses of dark matter subhalos using stellar streams is a key scientific objective of the Via project, currently planning a 5 year radial velocity (RV) survey of stellar stream members to precisions of 0.1 km s^{-1} .

Email: anya.phillips@cfa.harvard.edu

The orbital motions of binary stars present a source of intrinsic velocity dispersion in stellar streams that can obscure the signatures of a subhalo impact. In analyses searching for subhalo impacts, binaries must be identified through RV scatter over multiple observations to be rejected from the sample. If a Keplerian orbit can be fit to its RV curve (generally true of binaries with orbital periods shorter than the observing time baseline), the systemic velocity of a binary system can be included in further dark matter-related analysis.

The systemic velocity of a binary detected based on RV scatter is relatively straightforward to recover with reasonable accuracy by simply taking the average of the RV measurements. However, recovering the full set orbital parameters lends insight into the actual binary demographics and underlying binary evolution (e.g., T. Mazeh & D. Goldberg 1992; S. Meibom & R. D. Mathieu 2005; M. H. Pinsonneault & K. Z. Stanek 2006; J. C. Lurie et al. 2017; A. M. Geller et al. 2021; E. M. Leiner et al. 2022). While the presence of binaries in globular clusters and their role in these systems’ dynamical evolution has been studied at length both observationally (e.g., V. Sommariva et al. 2009; B. Giesers et al. 2019; F. Wragg et al. 2024) and with N-body simulations (e.g., P. Hut et al. 1992; E. P. Rubenstein & C. D. Bailyn 1997; M. D. Albrow et al. 2001; J. R. Hurley et al. 2007; N. Ivanova et al. 2005), whether binary systems typically

survive the process of their host cluster’s dynamical relaxation and enter its tidal tails as bound systems is not known. With time-series RV measurements spanning up to 5 years, the Via survey stands to shed first light on the demographics of short-period binaries in globular cluster streams.

We can obtain, by an iterative procedure, a definitive least-squares fit to radial velocity data yielding best-fitting values of the Keplerian orbital parameters of binary stars (e.g., N. Katoh et al. 2013). More sophisticated approaches involve stochastic sampling techniques that deliver posterior probability distributions for the parameters, allowing for better statistical modeling at the population level. The current tool of choice in the field is TheJoker, presented in A. M. Price-Whelan et al. (2017). Their rejection sampling scheme is built to produce correct samplings even in cases where the posterior is highly multi-modal due to sparse RV curves. However, any Monte Carlo sampler is limited by high computational costs, a potential problem for the thousands of stream members that Via will observe.

Machine learning approaches stand to alleviate the issue of computational expense as long as they still offer a means to quantify uncertainty. One option are variational models (I. Oleksiienko et al. 2023), in which a neural network is trained to output the mean and standard deviation of a normal distribution that approximates the posterior of the model parameters. This approach has two issues. First, that the shape of the posterior is inflexible; for sparsely sampled RV curves, a Gaussian will not capture the true shape of the distribution. Second, variational networks are optimized using the evidence lower bound (ELBO), which requires computing the likelihood of each training data point at each training iteration. At large scales, this is expensive for binary orbits, since computing model predictions requires numerically solving a transcendental equation (see Section 2).

Because RV curves are easy to forward model from simulated orbital parameters, this problem is well-suited to simulation-based inference (SBI). In this work, we present ANKLES², an SBI model trained by optimizing on a proposed posterior of ground-truth simulated data (eliminating the need to compute the likelihood), to output full posteriors (which are flexible in shape and not limited to Gaussians) generated by a normalizing flow. In Section 2 we present the mathematical formalism for Keplerian orbits and describe the generation of training data from a realistic population of binaries. In Section 3, we train a baseline model and discuss its per-

formance. In Section 4 we describe the architecture and training for the full model, before demonstrating its performance in Section 5. We conclude and discuss future improvements to ANKLES in Section 6.

2. DATA GENERATION

For a two-body system, the radial velocity v at time t is given by

$$v(t; \theta) = v_0 + K[\cos(\omega + f) + e \cos(\omega)], \quad (1)$$

where θ represents the model parameters, i.e., the systemic velocity v_0 , velocity semi-amplitude K , argument of periapsis ω , eccentricity e , and the true anomaly f given by

$$\cos(f) = \frac{\cos E - e}{1 - e \cos E}. \quad (2)$$

The eccentric anomaly E must be solved for using the mean anomaly M from Kepler’s equation:

$$M = \frac{2\pi t}{P} - \phi_0 \quad (3)$$

($\phi_0 = \frac{2\pi T_p}{P}$, where T_p is the time of periastron)

$$M = E - e \sin E. \quad (4)$$

Our models will seek the orbital parameters $v_0, K, \omega, \phi_0, e, P$, the systemic velocity, velocity semiamplitude, argument of periapsis, periastron phase term, orbital eccentricity, and orbital period, respectively.

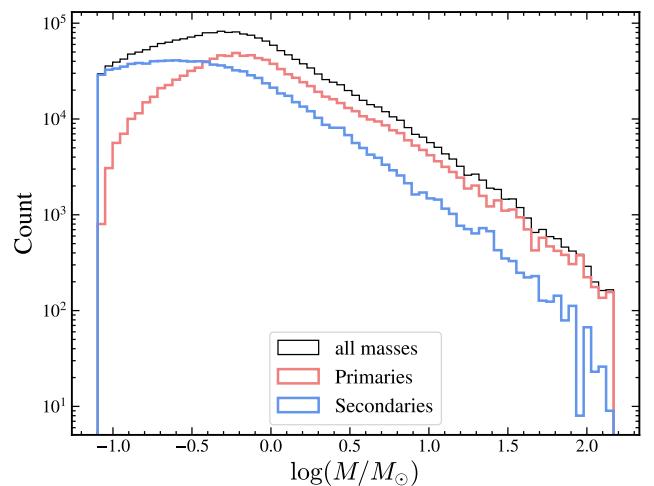


Figure 1. The mass function of the sample of all stars (black), primaries (pink), and secondaries (blue).

Rather than directly drawing the orbital parameters, we choose to first simulate a realistic population of binaries using the Python package COSMIC (K. Breivik

² Amortized Normalizing flow for Keplerian orbitL Estimation with SBI; <https://github.com/anya-m-phillips/ANKLES>

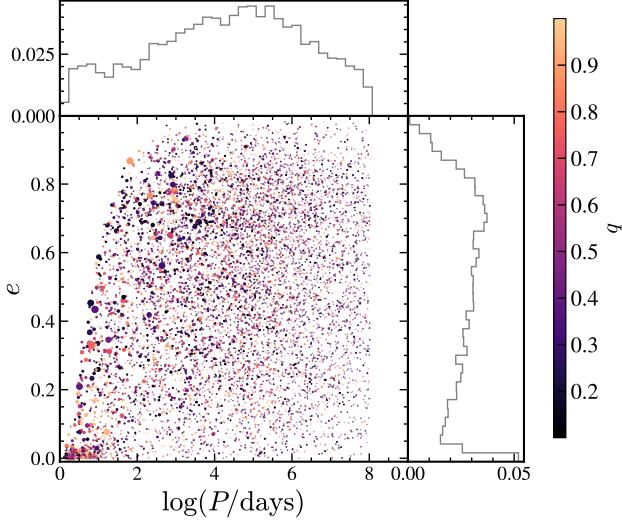


Figure 2. Distribution of a random subset of the sample in orbital eccentricity and period. The points are colored by binary mass ratio and sized in proportion to the systems total mass.

et al. 2020) before computing their orbital parameters as training data. COSMIC generates populations of both single and binary stars, where the binary fraction can be adjusted to depend on the mass of the primary. By using the subset of binaries returned by the code to generate training data, we realistically sample relative population sizes of binaries of varying stellar types. We assume a P. Kroupa (2001) initial mass function and draw binary mass ratios, orbital periods, and eccentricities from the covariant and mass-dependent distributions inferred by M. Moe & R. Di Stefano (2017). This assumes a binary fraction of $\sim 41\%$ for solar-type primaries, increasing to 57% at $M_1 = 3M_\odot$ and 80% at $M_1 = 6M_\odot$. Below $0.8M_\odot$, the binary fraction is assumed to decrease linearly with $\log M_1$, from 40% at $0.8M_\odot$ to 0 at $0.08M_\odot$. We sample 10^6 binaries with COSMIC.

Figure 1 shows the distributions in primary mass, secondary mass, and the full mass function. The deviations in power law slope from the standard P. Kroupa (2001) IMF come from the varying probabilities of forming in a binary system as a function of stellar mass. Figure 2 shows the distribution of a random subset of 1000 binaries in the sample in eccentricity and orbital period, with points colored by binary mass ratio $q = M_2/M_1$ and sized in proportion to the system's total mass. The eccentricities are drawn from the modified thermal distribution in M. Moe & R. Di Stefano (2017), where short-period binaries are required to have circularized orbits. The orbital periods roughly follow a log-normal distribution centered on $\sim 10^5$ days. Figure 3 shows the distribution of the sample in binary mass ratio. The sample

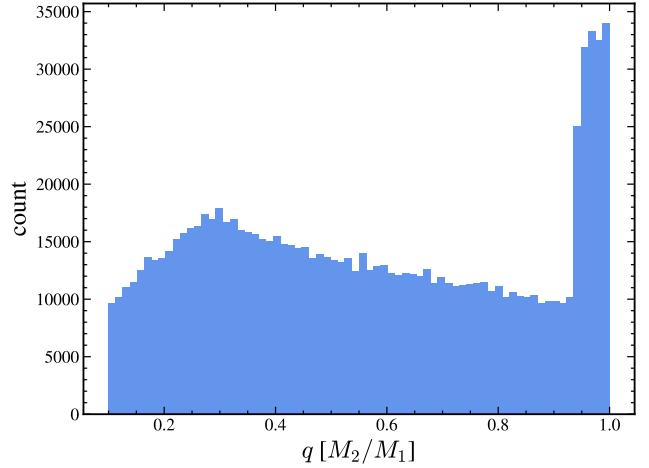


Figure 3. Distribution of the sample in binary mass ratio q .

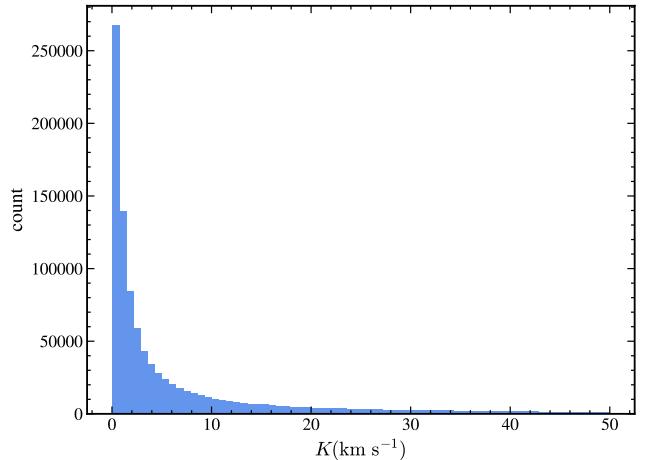


Figure 4. Distribution of the sample in velocity semiamplitude K .

follows the modified double power law distribution from M. Moe & R. Di Stefano (2017), with the empirically observed overabundance of twin binaries ($q = 1$).

We draw inclinations i from an isotropic distribution (uniform in $\cos i$), then compute the velocity semiamplitudes K as

$$K = \frac{M_2 \sin i}{\sqrt{1 - e^2}} \left(\frac{2\pi G}{P(M_1 + M_1)} \right)^{1/3}, \quad (5)$$

which yields the distribution in K shown in Figure 4. We draw arguments of periapsis ω from a uniform distribution $\mathcal{U}(0, 2\pi)$, and periastron phase terms ϕ_0 from $\mathcal{U}(0, 1)$. We draw systemic velocities v_0 from a Gaussian, $\mathcal{N}(\mu = 0, \sigma = 30 \text{ km s}^{-1})$, following A. M. Price-Whelan et al. (2017).

For the data features, we sample a time baseline of 5 years by drawing successive gaps between observations

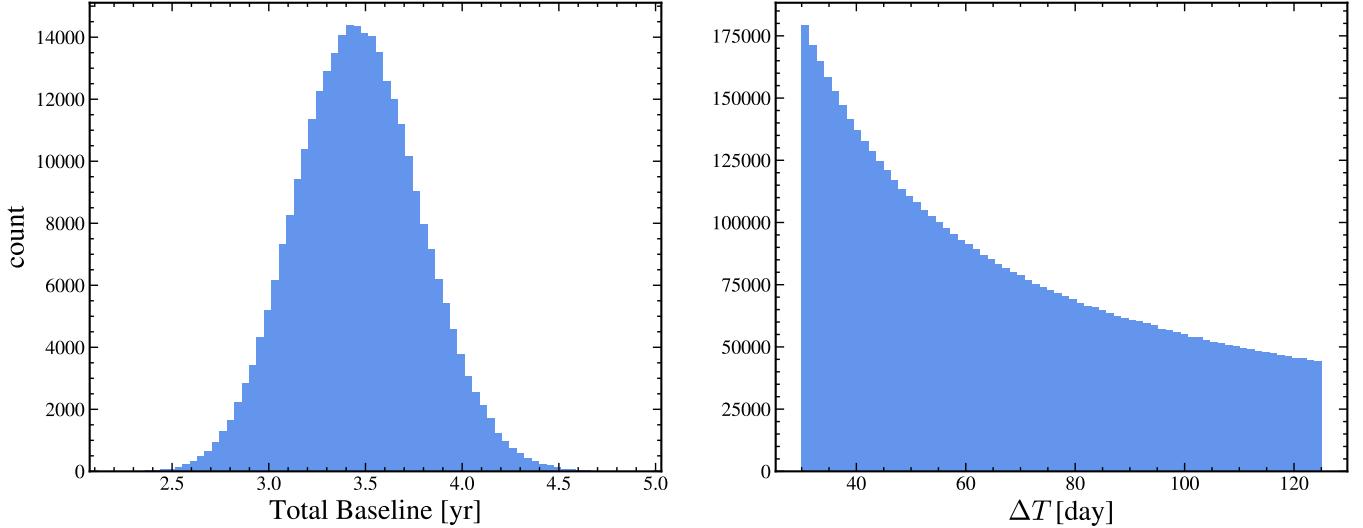


Figure 5. (Left) the distribution in total baseline for each binary in our sample. (Right) the distribution in gaps between observations generated.

from a log-uniform distribution spanning 30 to 125 days. We generate 20 observation epochs for each binary. Figure 5 shows the resulting distribution in total baselines, as well as the log-uniform distribution of successive gaps between visits. The distribution in total baselines is peaked around 3.5 years ($\sim 10^3$ days), meaning that, intuitively, our models performance will likely degrade with increasing true orbital period where the full shape of the phased RV curve is not available.

With observing times starting from zero and the orbital parameters for each system, we are able to compute the corresponding radial velocities by numerically solving Kepler's equation (Equations 3 and 4) at each time, computing the true anomaly (Equation 2), and substituting into Equation 1. After generating the radial velocities, we add Gaussian noise with $\sigma = 0.1 \text{ km s}^{-1}$ to mimic Via's target RV precision. Figure 6 shows 10 randomly selected and phase-folded RV curves from this process (where the orbital phase $\phi = (t - t_0/P) - \text{int}(t - t_0/P)$), including the model using the underlying orbital parameters.

3. TRAINING A BASELINE

We perform an 80/20 train/test split and train a basic multi-layer perceptron (MLP) as a baseline for comparison with our final model in Section 4. As inputs, we separately normalize the radial velocities and observation times using min-max scaling (so that the RVs and observation times each span the range 0–1). The normalization scheme allows the network to be tuned finely to estimate the nonlinear parameters (ω, ϕ_0, e, P), but it removes all physical information from the inputs about the systemic velocity v_0 and velocity amplitude

K . To remedy this, we pass two additional features as inputs to the network: the average of the radial velocities v_{avg} (tracing the systemic velocity) and the estimated semiamplitude based on the minima and maxima of the RV curve, $K_{\text{est}} = (v_{\text{max}} - v_{\text{min}})/2$. We normalize the training outputs using min-max scaling so that each orbital parameter in the training set lies in the range 0–1. This means that the network outputs normalized parameters, which must always be un-normalized using the minima and maxima from the original training data. Because the orbital periods are generated based on a \sim log-normal distribution, we choose to train the network to estimate the log periods.

The baseline MLP has one hidden layer with 20 nodes and a leaky ReLU activation function. We train it for 1000 epochs using the mean-squared error as the objective function, PyTorch's implementation of the ‘adam’ optimizer, and a learning rate of 10^{-5} . We find that the improvement begins to plateau after only about 200 epochs, but there is no evidence of overfitting based on the successive losses of the test data. Figure 7 shows the test dataset in the predicted model parameters by the baseline model versus their true values and color-coded by the true value of the orbital period. As expected, the systemic velocity and semiamplitude are recovered with high accuracy, given that we pass reliable estimators of their values to the network as inputs. The argument of periaxis ω shows a reasonable correspondence between the predicted and true values for intermediate ground truth values, with degraded performance near 0 and 2π , but this is not concerning given the cyclical nature of this parameter, where $\omega = 0$ and $\omega = 2\pi$ yield identical RV predictions. The periastron phase and eccentric-

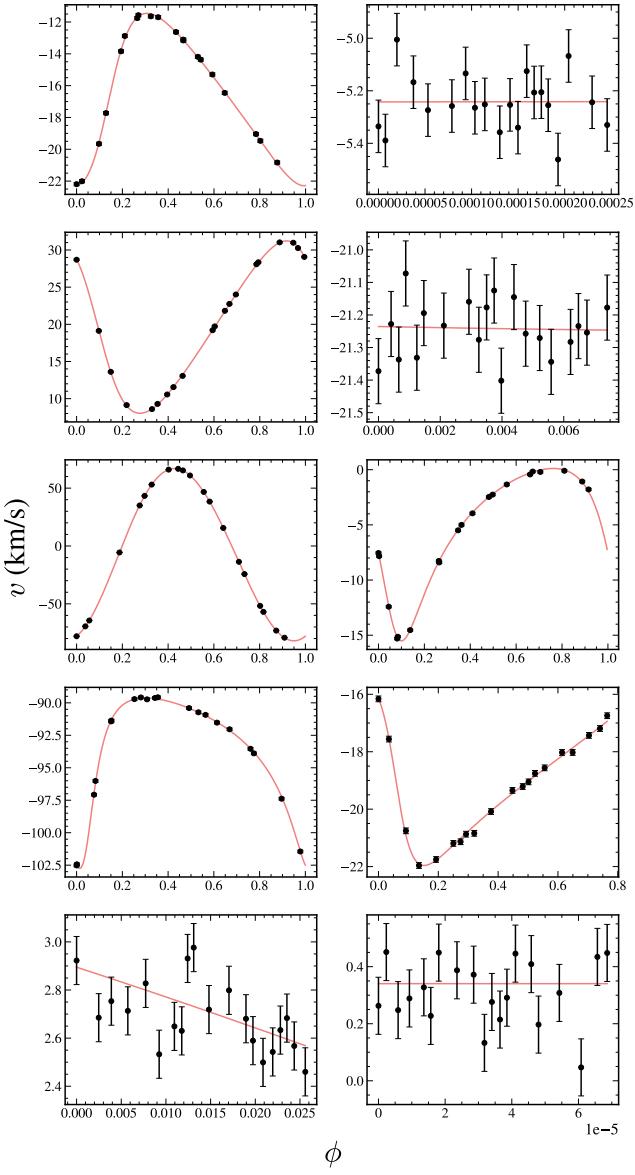


Figure 6. Ten phased RV curves randomly sampled from those generated for training data, showing the model using the underlying orbital parameters in pink. For systems with orbital periods significantly longer than the visit time baseline, the phase coverage is minimal.

ity seem particularly difficult for the model to estimate, meaning that accurately characterizing their uncertainties will be important in our more Bayesian model. The log-periods seem to be predicted with reasonable accuracy up to periods $\lesssim 10^{4.5}$ days. The model’s main failure mode is visible in the bar-like features of the ω , ϕ_0 , and e panels, where it always predicts the average value of the training sample. We see from the colormap that this feature corresponds to the long-period points where the model cannot distinguish orbits with different

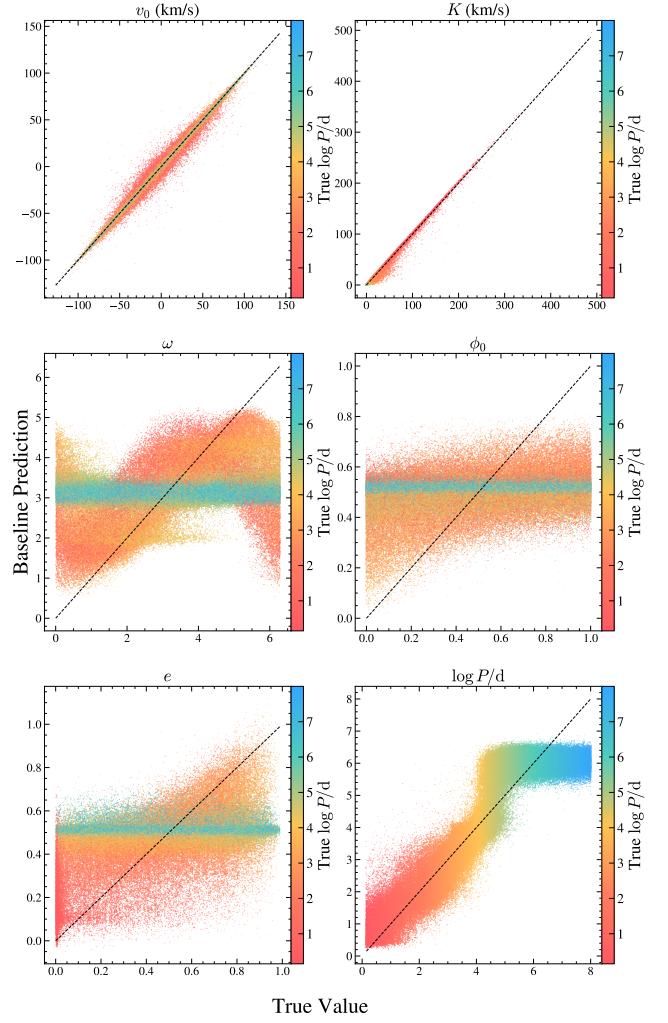


Figure 7. Predicted values of each orbital parameter using the baseline MLP versus their true values. Dark diagonal lines show one-to-one correspondences. The colormap corresponds to the true value of the period.

parameters due to insufficient coverage over the orbital phase space.

4. TRAINING THE FULL MODEL

We use the package `sbi` to build our full model. The package requires a prior object in order to efficiently sample the distributions it produces, so we build a custom prior by smoothing our training labels with a Kernel Density Estimation from `scikit-learn`. The model itself consists of a normalizing flow, where a simple Gaussian is morphed into an approximation of the true posterior through a series of invertible transformations f_k with easily-computable Jacobian determinants. The full expression for the final posterior is given by

$$\log p(z_K) = \log p_0(x_0) - \sum_{k=1}^K \log \left| \frac{\partial f_k}{\partial z_{k-1}} \right|. \quad (6)$$

For the functions f_k , we choose the “Masked Autoregressive Flow” (MAF). In these transformations, the input and output are both vectors in \mathbb{R}^6 (the dimension of our vector of 6 orbital parameters), and each output dimension y_i is computed as

$$y_i = \mu_i(x_{<i}) + \sigma_i(x_{<i}) \cdot x_i, \quad (7)$$

where $x_{<i}$ denotes all input dimensions before i . μ_i and σ_i are neural networks that depend only on $x_{<i}$ (for input vector x). Each dimension is transformed sequentially, and this “autoregressive” structure ensures a triangular Jacobian, so that the log-determinant is cheap to compute:

$$\log \left| \det \frac{\partial y}{\partial x} \right| = \sum_i \log \sigma_i(x_{<i}). \quad (8)$$

We use 10 flow layers, each involving neural networks with 20 neurons (`nblocks=10; nhiddens=20` in `sbi`). The networks are optimized using gradient descent with the objective that output posterior q_ϕ is maximized at the ground truth values of the model parameters given the data,

$$\max_\phi \sum_i \log(\theta_i|x_i), \quad (9)$$

which should be true if q_ϕ well approximates the true posterior. Given the computational expense of training, we opt to train with only a subset of the 10^6 points generated in Section 2. We randomly select 50,000 points and perform an 80/20 train/test split for the SBI model.

5. MODEL PERFORMANCE

Figure 8 shows an example visualization of 40,000 samples of the output posterior for one of the systems in the test set. Solid lines and square markers denote the ground truth for this system. Encouragingly, the model has captured non-gaussian shapes, and in most cases the true value overlaps the contoured region (which by default encloses the 3σ credible region) of the estimated posterior.

After drawing 1000 samples from every posterior in the test sample, we plot the median values of each parameter versus the ground truth in Figure 9. Here, the color corresponds to a proxy for the uncertainty in the median estimate, the 16th–84th percentile range of the posterior. We see again that v_0 and K are well-estimated, as with the baseline. The bands of failed estimates for the parameters of systems with long periods are still present in the panels for ω , ϕ_0 , and e , but in particular for ω and e , we note that these systems have correspondingly high uncertainty estimates, with points

that are better estimated having lower model uncertainties. The model seems best able to characterize high-eccentricity systems, and less able to distinguish lower eccentricities. The period estimates correspond well to the ground truths, with failed estimates for long-period systems having high uncertainties. The ϕ_0 estimates and uncertainties are the least well-behaved, where the uncertainty seems to correspond mostly to the estimated value of ϕ_0 . For small median ϕ_0 , the uncertainty is low and for large ϕ_0 , the uncertainty is high, with the “failed” long-period systems having only intermediate uncertainties.

To assess whether the model yields the “true” posterior, we show a predictive probability plot in Figure 10. This shows the cumulative distribution function in model parameters ($P(\leq \theta)$ for each parameter θ), for the model estimates compared to the ground truth. We find very good agreement between the two³, though we note slightly overconfident curves for e and ω , and slightly biased curves for ϕ_0 and v_0 . From this figure, we determine that our model sufficiently recovers the “true” posterior on the model parameters.

As a final test of the model, we show in Figure 11 an example RV curve from the test dataset with 100 sampled sets of model parameters from the posterior output by ANKLES. The model from the true parameters is shown as a black dashed line and the model from the median of the posterior samples is shown as a pink dashed-dotted line. This example was chosen primarily because the period is such that several orbits fit within the observing baseline (but not so many orbits as to crowd the plot before the RV curve is phase-folded). While the median from the posterior samples does not perfectly match the ground truth, it very roughly captures the true period, eccentricity, as well as the systemic velocity and velocity amplitude.

6. CONCLUSION

We have presented ANKLES, an SBI model to compute the posterior distribution for orbital parameters of binary stars based on their radial velocity curves. We showed that while point estimates taken at the median of the posterior compared to ground truth posteriors behave similarly to those estimated by a vanilla MLP, the parameter uncertainties correctly indicate where the predictions deviate from the ground truths.

ANKLES is available as a GitHub repository at <https://github.com/anya-m-phillips/ANKLES>, which

³ If you attended the AY205 poster session, you may recall a PP plot that looked *much* worse. This was due to a bug in the code that computed the ranks for each test data sample!

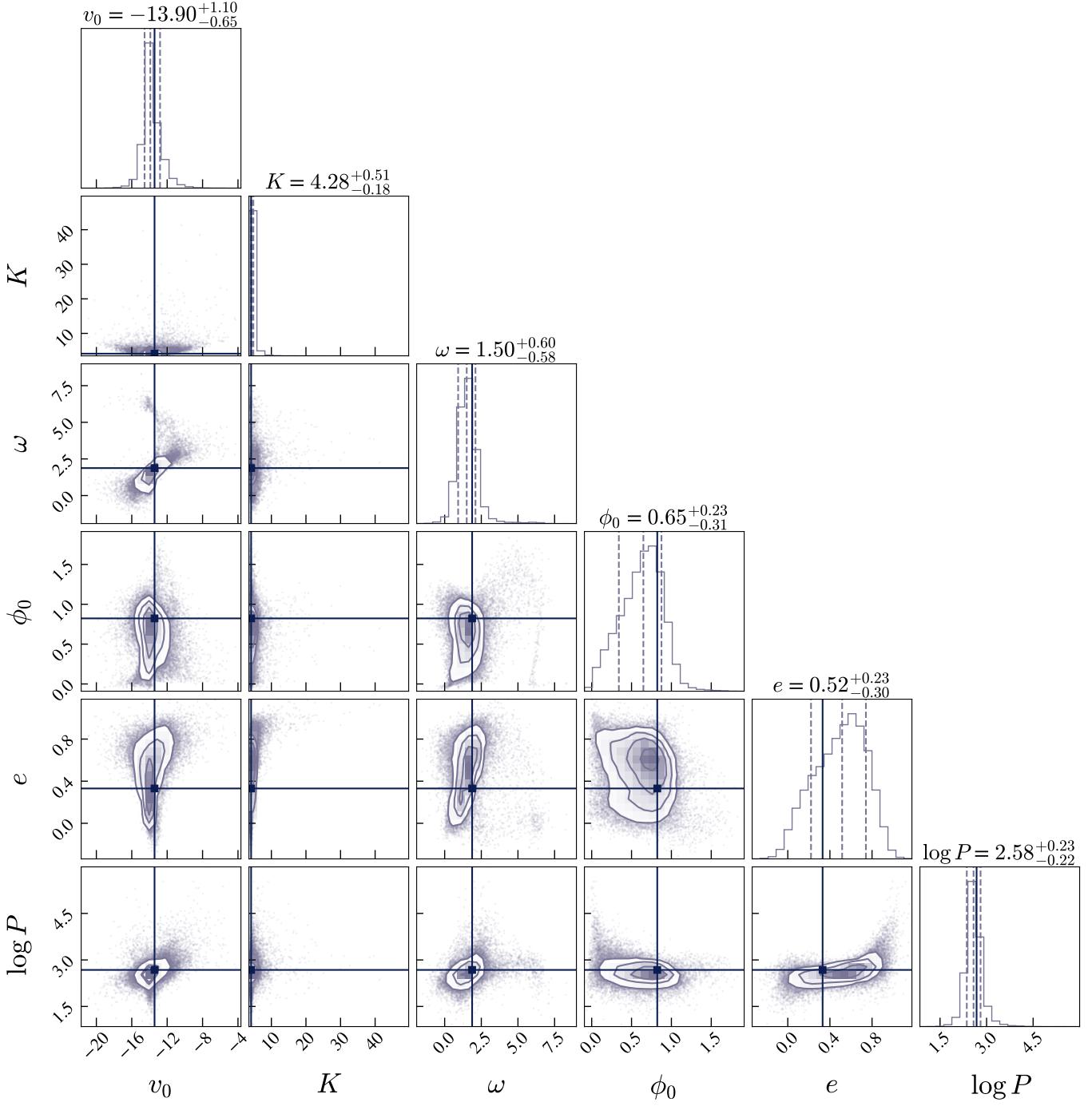


Figure 8. Corner plot showing a 40,000 posterior samples for one system in the test set. The ground truth values are shown with dark lines/square points.

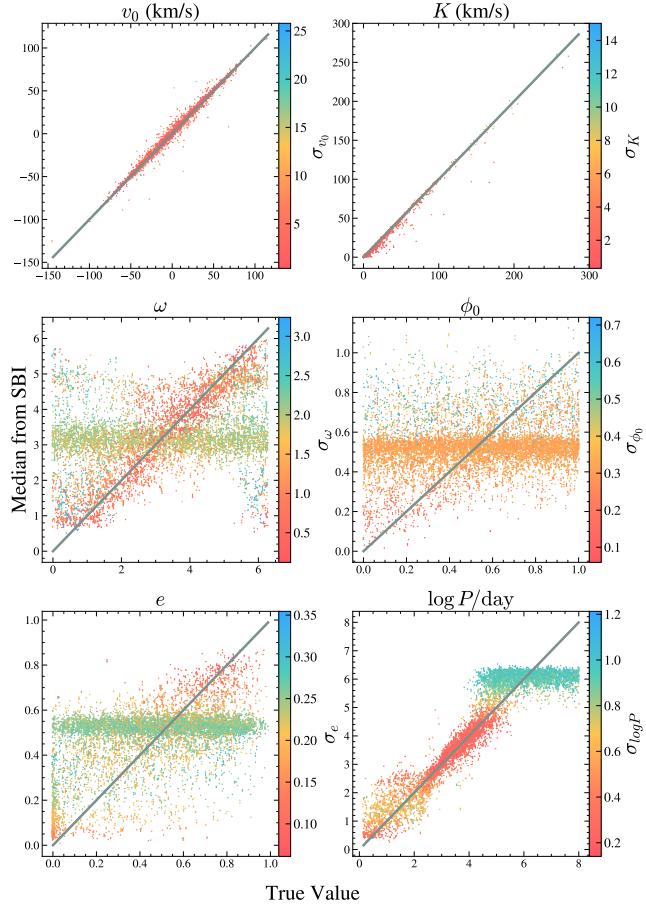


Figure 9. After sampling the posteriors for the test dataset, the median values in each parameter as predicted by the ANKLES versus the ground truth value. The colormap corresponds to the 16th–84th percentile range of the posterior, a proxy for the width of the distribution and uncertainty in the median estimate.

includes the notebooks used to generate training data and train all models in this report. It also includes a script to quickly call the SBI model and sample posteriors given input RV curves, as well as generate synthetic RV curves for testing using Kepler’s equation and the time-sampling cadence presented in Section 2.

While the results of this work are exciting (in particular, Figure 10), there remain numerous improvements to ANKLES which will make it better suited to working with upcoming Via data. First, we have generated “best case scenario” RV curves with 20 visits, but Via’s actual observing strategy is more likely to visit stars only 3 times (twice near the beginning of the survey, and once at the end). While this strategy has been shown to allow for *identification* of the binaries contributing significant RV dispersion to stellar streams (Via Science Book, in preparation), the orbital parameters are significantly more difficult to recover. A version of ANKLES trained

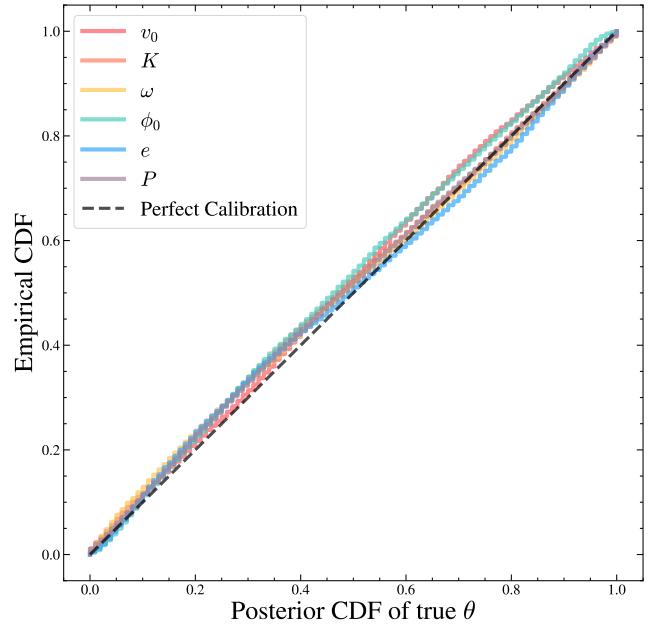


Figure 10. Predictive probability calibration plot showing the empirical CDF compared to the CDF for the ground truth parameters. Different colored lines denote the results for each model parameter and a black dashed line shows the perfect agreement.

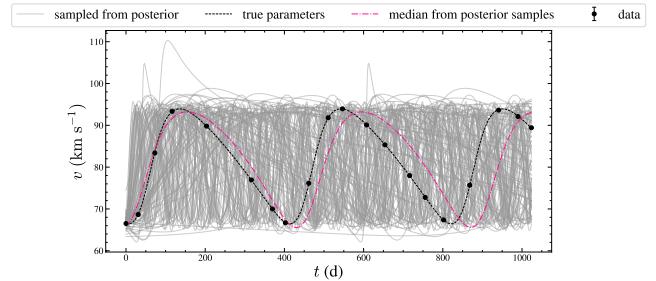


Figure 11. An example RV curve from the test data set with the model from true parameters (black dashed) and 100 models sampled from the posterior (gray solid) with the median from the posterior samples (pink dashed-dotted).

on 3-visit RV curves would need to be able to reproduce the highly multi-modal posteriors in orbital parameters in order to yield viable statistical information in studies of the binaries in Via data.

Our noise model assumed a blanket 0.1 km s⁻¹ error on all RV measurements. However, Via RV precision will be magnitude dependent, and will further depend on the surface convection properties of the observed stars, and their surface gravities due to the resulting gravitational redshift of photons arriving at the spectrograph. A more complete noise model must account for these added sources of RV jitter and bias, which could be accomplished using the primary masses and stellar types

generated by **COSMIC** to predict surface gravities and jitter parameters.

We have shown that **ANKLES** unsurprisingly fails to recover correct orbital parameters as the medians of the posteriors for systems with orbital periods significantly longer than the survey duration. It may therefore be valuable to retrain a model only for shorter-period systems, so that the training better prioritizes good solutions in regimes where it has a chance to make correct predictions.

Finally, there are a few more feature engineering choices one could make to improve **ANKLES**' performance. First, training on $\sin \omega$ and $\cos \omega$ would remove the oscillatory behavior seen in Figures 7 and 9. Further, since the input distribution of semiamplitudes K is more logarithmic than linear (Figure 4), training a model in $\log K$ as we did with $\log P$ might make more sense.

Software: **COSMIC** (K. Breivik et al. 2020), PyTorch (A. Paszke et al. 2019), SBI (A. Tejero-Cantero et al. 2020)

REFERENCES

- Albrow, M. D., Gilliland, R. L., Brown, T. M., et al. 2001, ApJ, 559, 1060, doi: [10.1086/322353](https://doi.org/10.1086/322353)
- Breivik, K., Coughlin, S., Zevin, M., et al. 2020, The Astrophysical Journal, 898, 71, doi: [10.3847/1538-4357/ab9d85](https://doi.org/10.3847/1538-4357/ab9d85)
- Diemand, J., Kuhlen, M., & Madau, P. 2007, ApJ, 657, 262, doi: [10.1086/510736](https://doi.org/10.1086/510736)
- Erkal, D., & Belokurov, V. 2015, MNRAS, 454, 3542, doi: [10.1093/mnras/stv2122](https://doi.org/10.1093/mnras/stv2122)
- Geller, A. M., Mathieu, R. D., Latham, D. W., et al. 2021, AJ, 161, 190, doi: [10.3847/1538-3881/abdd23](https://doi.org/10.3847/1538-3881/abdd23)
- Giesers, B., Kamann, S., Dreizler, S., et al. 2019, A&A, 632, A3, doi: [10.1051/0004-6361/201936203](https://doi.org/10.1051/0004-6361/201936203)
- Hurley, J. R., Aarseth, S. J., & Shara, M. M. 2007, ApJ, 665, 707, doi: [10.1086/517879](https://doi.org/10.1086/517879)
- Hut, P., McMillan, S., Goodman, J., et al. 1992, PASP, 104, 981, doi: [10.1086/133085](https://doi.org/10.1086/133085)
- Ivanova, N., Belczynski, K., Fregeau, J. M., & Rasio, F. A. 2005, MNRAS, 358, 572, doi: [10.1111/j.1365-2966.2005.08804.x](https://doi.org/10.1111/j.1365-2966.2005.08804.x)
- Katoh, N., Itoh, Y., Toyota, E., & Sato, B. 2013, AJ, 145, 41, doi: [10.1088/0004-6256/145/2/41](https://doi.org/10.1088/0004-6256/145/2/41)
- Klypin, A., Kravtsov, A. V., Valenzuela, O., & Prada, F. 1999, ApJ, 522, 82, doi: [10.1086/307643](https://doi.org/10.1086/307643)
- Kroupa, P. 2001, MNRAS, 322, 231, doi: [10.1046/j.1365-8711.2001.04022.x](https://doi.org/10.1046/j.1365-8711.2001.04022.x)
- Leiner, E. M., Geller, A. M., Gully-Santiago, M. A., Gosnell, N. M., & Tofflemire, B. M. 2022, ApJ, 927, 222, doi: [10.3847/1538-4357/ac53b1](https://doi.org/10.3847/1538-4357/ac53b1)
- Lurie, J. C., Vyhmeister, K., Hawley, S. L., et al. 2017, AJ, 154, 250, doi: [10.3847/1538-3881/aa974d](https://doi.org/10.3847/1538-3881/aa974d)
- Mazeh, T., & Goldberg, D. 1992, ApJ, 394, 592, doi: [10.1086/171611](https://doi.org/10.1086/171611)
- Meibom, S., & Mathieu, R. D. 2005, ApJ, 620, 970, doi: [10.1086/427082](https://doi.org/10.1086/427082)
- Moe, M., & Di Stefano, R. 2017, ApJS, 230, 15, doi: [10.3847/1538-4365/aa6fb6](https://doi.org/10.3847/1538-4365/aa6fb6)
- Moore, B., Ghigna, S., Governato, F., et al. 1999, ApJL, 524, L19, doi: [10.1086/312287](https://doi.org/10.1086/312287)
- Oleksiienko, I., Tran, D. T., & Iosifidis, A. 2023, Procedia Computer Science, 222, 104, doi: [10.1016/j.procs.2023.08.148](https://doi.org/10.1016/j.procs.2023.08.148)
- Paszke, A., Gross, S., Massa, F., et al. 2019, in Proceedings of the 33rd International Conference on Neural Information Processing Systems No. 721 (Red Hook, NY, USA: Curran Associates Inc.), 8026–8037
- Pinsonneault, M. H., & Stanek, K. Z. 2006, ApJL, 639, L67, doi: [10.1086/502799](https://doi.org/10.1086/502799)
- Price-Whelan, A. M., Hogg, D. W., Foreman-Mackey, D., & Rix, H.-W. 2017, The Astrophysical Journal, 837, 20, doi: [10.3847/1538-4357/aa5e50](https://doi.org/10.3847/1538-4357/aa5e50)
- Rubenstein, E. P., & Bailyn, C. D. 1997, ApJ, 474, 701, doi: [10.1086/303498](https://doi.org/10.1086/303498)
- Sommariva, V., Piotto, G., Rejkuba, M., et al. 2009, A&A, 493, 947, doi: [10.1051/0004-6361:200810696](https://doi.org/10.1051/0004-6361:200810696)
- Springel, V., Wang, J., Vogelsberger, M., et al. 2008, MNRAS, 391, 1685, doi: [10.1111/j.1365-2966.2008.14066.x](https://doi.org/10.1111/j.1365-2966.2008.14066.x)
- Tejero-Cantero, A., Boelts, J., Deistler, M., et al. 2020, Journal of Open Source Software, 5, 2505, doi: [10.21105/joss.02505](https://doi.org/10.21105/joss.02505)
- Wragg, F., Kamann, S., Saracino, S., et al. 2024, MNRAS, 535, 781, doi: [10.1093/mnras/stae2333](https://doi.org/10.1093/mnras/stae2333)
- Yoon, J. H., Johnston, K. V., & Hogg, D. W. 2011, ApJ, 731, 58, doi: [10.1088/0004-637X/731/1/58](https://doi.org/10.1088/0004-637X/731/1/58)