Astronomy and Interpretability: Random Forests Learn the Inverse Square Law

Sarthak Arora
U.C. Berkeley
sarthakarora@berkeley.edu

Naveen Durvasula
U.C. Berkeley
ndurvasula@berkeley.edu

Haydn Gwyn
U.C. Berkeley
haydngwyn@berkeley.edu

Oskar Hurst
U.C. Berkeley
oskar.hurst@berkeley.edu

Abstract

We demonstrate the ability of a random forest model to learn Newton's inverse-square law, particularly within the context of orbital mechanics, via the use of multiple techniques from model explainability literature. We train a random forest to predict the first finite difference of a planet's velocity based on the planet's prior position relative to Earth and its distance to the Sun. Shapley values of these features demonstrate the relative significance of the planet's distance to the Sun and exhibit a linear relationship with the inverse square of said difference, validating the recovering the inverse-square law up to a constant of proportionality. t-distributed stochastic neighbor embedding (TSNE) reveals the topological structure of the feature space for each planet examined, yielding the expected 1-dimensional manifold in all cases except for one. All code to reproduce experiments can be found at `https://github.com/ndurvasula/TeamOskarProjectS`.

1 Introduction

Over several millennia, various civilizations have developed their own mythologies as they relate to heavenly bodies such as the sun, the moon, or potentially other planets. The ancient Greeks acknowledged Zeus as the god of the sky and attribute each astronomical body to a different, less powerful figure (they understood the movement of the Sun as a result of the god Helios riding his chariot from the east to the west, for example). The Shinto Japanese attribute the (first) event of a solar eclipse to a conflict between the sun goddess Amaterasu and her brother Susanoo. The Norse foresee the devouring of the sun and moon by the Fenris wolf at Ragnarok, signifying an elevated conception of significance of these two bodies. The Chinese identify each year with one of the twelve members of the zodiac. In *all* cases, however, the goal of an astronomical mythos is to *explain* the behavior of astronomical bodies, be it the sun, the moon, the stars, or the planets.

Today, we understand the movement of astronomical objects in terms of Kepler's Laws. Kepler's first law states that the orbit of a planet is given by an ellipse, with the Sun at one of the two foci. Formally, we have that the distance r from any planet to the Sun takes the form

$$\|\mathbf{r}\| = \frac{p}{1 + \epsilon \cos\theta}$$

where $\mathbf{r}$ is the vector from the planet to the sun, p denotes the semi-latus rectum, ϵ denotes the eccentricity of the ellipse, and θ denotes the angle from the planet to the Sun relative to the other focus. We illustrate these below in Figure 1.

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

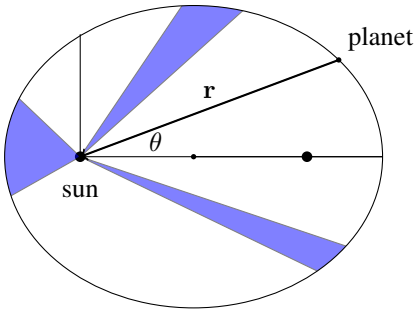

Figure 1: A diagram of the elliptical orbit (as per Kepler's first law) of a planet about the sun. We label θ and the vector $\mathbf{r}$ and show three regions of equal area swept out in equal times as per Kepler's second law. The vertical line segment through the sun is the semi latus rectum.

Kepler's second law states that angular momentum is conserved, or geometrically, that the line segment connecting the sun and an orbiting planet traces out equal areas in equal times (see Figure 1). For simplicity, we write

$$r(\theta) = \frac{A}{B + \cos\theta}$$

where r is the distance from an orbiting planet to the sun (this is equivalent to the above form under proper substitutions of constants). Then we have

$$A\dot{r} = \frac{A^2 \sin\theta}{(B+\cos\theta)^2}\dot{\theta} = \frac{A^2}{(B+\cos\theta)^2} \cdot \dot{\theta}\sin\theta = (r^2\dot{\theta})\sin\theta = C\sin\theta$$

Note that the area swept our by a planet's radius vector in a very short time period is well approximated by an isosceles triangle with legs of length $r(\theta)$ and included angle $d\theta = \dot{\theta}\, dt$. It follows that $r^2\dot{\theta}$ is constant, which validates the last equality. Now we have

$$1 = \sin^2\theta + \cos^2\theta = \frac{A^2}{C^2}\dot{r}^2 + (B - A/r)^2 = \frac{A^2}{C^2}\dot{r}^2 + B^2 - \frac{2A}{r} + \frac{A^2}{r^2}$$

Differentiating with respect to t gives

$$0 = \frac{2A^2}{C^2}\dot{r}\ddot{r} + \frac{2A}{r^2}\dot{r} - \frac{2A^2}{r^3}\dot{r}$$

Dividing by $\dot{r}$ and rearranging gives

$$\ddot{r} = \frac{C^2}{r^3} - \frac{C^2}{Ar^2}$$

Note also that

$$\frac{d^2}{dt^2} r\cos\theta = \frac{d}{dt}\dot{r}\cos\theta - r\dot{\theta}\sin\theta = \frac{d}{dt}\dot{r}\cos\theta - \frac{C}{r}\sin\theta = \ddot{r}\cos\theta - \frac{C^2}{r^3}\cos\theta = -\frac{C^2\cos\theta}{Ar^2}$$
$$\frac{d^2}{dt^2} r\sin\theta = \frac{d}{dt}\dot{r}\sin\theta + r\dot{\theta}\cos\theta = \frac{d}{dt}\dot{r}\sin\theta + \frac{C}{r}\cos\theta = \ddot{r}\sin\theta - \frac{C^2}{r^3}\sin\theta = -\frac{C^2\sin\theta}{Ar^2}$$

and hence the magnitude of the acceleration of a planet is proportional to

$$\sqrt{\left(\frac{d^2}{dt^2} r\cos\theta\right)^2 + \left(\frac{d^2}{dt^2} r\sin\theta\right)} = \frac{C^2}{Ar^2} \propto \frac{1}{r^2}$$

which is Newton's famous inverse-square law for the gravitational force.

Using tools from the model explainability literature, we aim to demonstrate that modern machine learning techniques have the capacity to learn these laws from observational data. We first give a brief overview of the techniques we will use.

1.1 Random Forests

Random forests consist of an ensemble of decision trees, where the features that each tree uses for prediction are generated by taking a random subset of the original features (this is generally referred to as *feature bagging* [Bre01]). As is typical of many ensemble learning algorithms, we average the predictions produced by the individual decision trees. This is why feature bagging is useful: it reduces correlation between the individual decision trees that we train, and hence the aggregation of these trees reduces the variance of our model. In particular, if we let $\hat{f}_{T_1}, \hat{f}_{T_2}, \ldots, \hat{f}_{T_n}$ be the prediction functions learned by each of the individual decision trees, then defining

$$\hat{f}_{agg} = \frac{1}{n} \sum_{i=1}^{n} \hat{f}_{T_i}$$

allows us to say

$$\mathrm{Var}[\hat{f}_{agg}] = \frac{1}{n^2} \sum_{i=1}^{n} \mathrm{Var}[\hat{f}_{T_i}]$$

which, assuming a roughly constant variance across the decision trees, yields a variance that decreases as n^{-1}. At the same time, the bias of the aggregate model is nothing but the average bias of each of the individual trees, and assuming again that these are roughly the same, we see that this aggregation approach does not affect bias.

Random forests have enjoyed success in a wide variety of applications, including ecology, genetics, and agriculture [CEJB$^+$07, SBB$^+$18, LLLW14]. Although random forests are highly versatile, they are not traditionally viewed as interpretable in the same way that standard tools based on linear models are as there is no equivalent of a "weight vector" that succinctly reports the relative importance of different features. This lack of explainability is a common feature to many modern function approximators, and is perhaps the central issue that we aim to address (in the particular context of astronomy) in this project.

1.2 Shapley Values

A modern technique known as Shapley Value Analysis [LL17b] provides a unified method for model explainability. The method has its roots in a subfield of mathematical economics known as mechanism design. In a seminal paper, Lloyd Shapley introduced a concept which is referred to today as a "Shapley Value" [Sha53]. This value is given by the average marginal contribution of a given player in an n-person game over all coalitions of players. In [LL17b], the task of prediction is thought of as an d-person game played amongst the d features. The SHAP value then computes the average marginal contribution of a feature in the prediction over all coalitions of features. We make this construction more explicit. Suppose we have learned a model $f : \mathbb{R}^d \to \mathbb{R}$, and we wish to explain its predictions on a dataset $\mathcal{X} \subset \mathbb{R}^d$. A *locally accurate additive feature attribution method* is a collection of functions $\phi_i : \mathbb{R}^d \to \mathbb{R}$ for $i \in [d]$ such that for all $\mathbf{x} \in \mathcal{X}$

$$f(\mathbf{x}) = \phi_0 + \sum_{i=1}^{d} \phi_i(\mathbf{x}) \mathbb{I}\left[x_i \text{ is known}\right]$$

The idea behind this model is to represent each prediction $f(\mathbf{x})$ as a linear combination of Boolean variables indicating whether or not the ith feature is known. The value of $\phi_i(\mathbf{x})$ is then easily interpretable as the impact of the ith feature on the model estimate $f(\mathbf{x})$. SHAP values, as proposed in [LL17b], define

$$\phi_i(\mathbf{x}) = \sum_{S \subset [d] \setminus \{i\}} \frac{|S|!(d - |S|! - 1)}{d!} \left| \mathrm{E}\left[f(\mathbf{x}) \mid \pi_{S \cup \{i\}}\right] - \mathrm{E}\left[f(\mathbf{x}) \mid \pi_S(\mathbf{x})\right] \right| \tag{1}$$

where $\pi_S : \mathbb{R}^d \to \mathbb{R}^{|S|}$ denotes the natural projection onto the indices given by $\mathbf{x}$. Equation 1 corresponds to the usual Shapley value formula introduced in the economics literature in [Sha53]. Using game-theoretic arguments, Lundberg et al. showed that SHAP values are the only *consistent* locally accurate additive feature attribution method. That is, it is the unique solution for the ϕ_i such

that if f were to place a larger importance on feature i at some point $\mathbf{x}$, $\phi_i(\mathbf{x})$ will never decrease. In fact, SHAP values are easily intepretable as conditional expectations. One may write

$$\phi_0 + \sum_{i=1}^{k} \phi_i(\mathbf{x}) = \mathrm{E}\left[f(\mathbf{x}) \mid x_1, \ldots, x_k\right]$$

In general, computing SHAP values is time consuming. As there are 2^d possible coalitions of features, computing the SHAP value of a given set of features takes exponential time. However, for the special case of tree ensembles, SHAP values can be computed in quadratic time using a recursive algorithm [LL17a]. As random forests belong to this class of predictors, SHAP values offer a locally accurate, consistent, and easily interpretable method for analyzing feature importance.

2 Method

We aim to show that a random forest model, given only observational data specifying a planet's position over time, can learn the inverse square law $\|\ddot{\mathbf{r}}\| \propto \frac{1}{r^2}$ described in Section 1. We generate data using the Python module `astropy`, which allows users to interface with the JPL Horizons On-Line Ephemeris System to get observations of planetary bodies over a specified interval of time [ART+13, APS+18]. We generate 50000 samples for the positions and velocities of the planets `Mercury`, `Venus`, `Mars`, `Jupiter`, and `Saturn`, which we denote by $\mathbf{p}_{\texttt{Mercury}}$, $\mathbf{p}_{\texttt{Venus}}$, $\mathbf{p}_{\texttt{Mars}}$, $\mathbf{p}_{\texttt{Jupiter}}$, and $\mathbf{p}_{\texttt{Saturn}}$ respectively. For each planet, we sampled over a period of time at least twice the length of its period. Table 1 gives the length of the observation period for each planet.

`Mercury`	`Venus`	`Mars`	`Jupiter`	`Saturn`
500	1000	1000	11000	22000

Table 1: Number of days that samples were gathered for each planet. Although the length of time for which we gather samples varies across planets, a total of 50000 samples were gathered for each planet.

We additionally generate $p_{\texttt{Sun}}$ for each respective time period. These positions are generated as Cartesian coordinates in the basis given by the International Celestial Reference System. The origin of this coordinate system is the barycenter of the Solar System, which is relatively close to the Sun. For any $\texttt{Planet} \in \{\texttt{Mercury}, \texttt{Venus}, \texttt{Mars}, \texttt{Jupiter}, \texttt{Saturn}\}$, we can compute the vector $\mathbf{r}_{\texttt{Planet}}$ (and similarly $\dot{\mathbf{r}}_{\texttt{Planet}}$) as described in 1 as

$$\mathbf{r}_{\texttt{Planet}}(t) = \mathbf{p}_{\texttt{Sun}}(t) - \mathbf{p}_{\texttt{Planet}}(t)$$

Using this data, we compute an approximate second difference

$$y_{\texttt{Planet}}(t) := \|\dot{\mathbf{r}}_{\texttt{Planet}}(t + \Delta t) - \dot{\mathbf{r}}_{\texttt{Planet}}(t)\| \approx \|\ddot{\mathbf{r}}_{\texttt{Planet}}\|$$

where Δt denotes the number of days per sample, which may be computed using the information in Table 1. We give the random forest the following information as input

$$\mathbf{x}_{\texttt{Planet}}(t) = \left(\alpha_{\texttt{Planet}}(t), \delta_{\texttt{Planet}}(t), \|\mathbf{p}_{\texttt{Earth}}(t) - \mathbf{p}_{\texttt{Planet}}(t)\|, \|\mathbf{r}_{\texttt{Planet}}(t)\|\right)$$

where $\alpha_{\texttt{Planet}}(t)$ denotes the right ascension of the planet, $\delta_{\texttt{Planet}}(t)$ denotes the declination of the planet, and $\|\mathbf{p}_{\texttt{Earth}}(t) - \mathbf{p}_{\texttt{Planet}}(t)\|$ denotes the distance from the planet to Earth. The right ascension and declination may be computed from the given Cartesian coordinates directly by using a spherical coordinate transformation. We did not include planets beyond `Saturn` in our analysis, as finite differences from the Ephemeris system become too inaccurate. Indeed, the finite difference data for Saturn is already inaccurate enough to produce substantial repercussions in our analysis, as we will see in Section 3. Observe that in this setup, we should have that

$$y_{\texttt{Planet}}(t) = \frac{C}{\|\mathbf{r}_{\texttt{Planet}}(t)\|^2}$$

by the inverse square law for some scaling constant C, and the other three features may only serve to contaminate prediction. We choose this setup as this information is similar to what early astronomers might have had access to. For each planet, we train a random forest $f_{\texttt{Planet}}$ on the dataset $(\mathbf{x}_{\texttt{Planet}}, y_{\texttt{Planet}})$ using the `scikit-learn` framework [PVG+11]. We first perform 3-fold randomized hyperparameter selection to tune the parameters `n_estimators`, `max_features`, `max_depth`,

`min_samples_split`, and `bootstrap` for 100 iterations. We then train the random forest using the optimized hyperparameters and verify that it performs well both quantitatively and qualitatively by computing the Bias and Variance of the estimator, along with a Gaussian Kernel Density Estimate of the residuals. We then compute the Shapley values of f to get the functions $(\phi_\alpha, \phi_\delta, \phi_E, \phi_S)$ corresponding to the features of $\mathbf{x}$ using the Python package `shap` [LEC+20]. If the random forest learned $y_{\texttt{Planet}}$ correctly, we should have that

$$\phi_S(\|\mathbf{r}_{\texttt{Planet}}\|) + \phi_0 \approx y_{\texttt{Planet}} \propto \frac{1}{\|\mathbf{r}_{\texttt{Planet}}\|^2}$$

and $\phi_\alpha, \phi_\delta, \phi_E \approx 0$. We verify the former by showing that $\frac{1}{\|\mathbf{r}_{\texttt{Planet}}\|^2}$ and $\phi_S(\|\mathbf{r}_{\texttt{Planet}}\|)$ are affinely related. The latter may be verified by comparing the empirical distributions of $(\phi_\alpha, \phi_\delta, \phi_E, \phi_S)$ over a random choice of $\mathbf{x}$. We further analyze the contamination of the three extraneous features on the Shapley values by computing a 2-dimensional t-distributed Stochastic Neighbor Embedding (TSNE) of the point cloud

$$\mathcal{M} := \{(\phi_\alpha(\mathbf{x}(t)), \phi_\delta(\mathbf{x}(t)), \phi_E(\mathbf{x}(t)), \phi_S(\mathbf{x}(t)))\}$$

first introduced in [MH08]. In theory, $TSNE(\mathcal{M}) \subset \mathbb{R}^2$ should appear as a 1-dimensional manifold, as all coordinates, except that corresponding to ϕ_S, should be zero. However, if the random forest splits on the other features, we may instead see clusters of points corresponding to groups of $\mathbf{x}(t)$ that are treated similarly by the random forest.

3 Results

We first analyze how well our random forests fit $y_{\texttt{Planet}}$ in terms of bias and variance. Table 2 demonstrates that the random forest predicts the targets $y_{\texttt{Planet}}$ between 5 to 8 orders of magnitude better than the mean $\mathrm{E}[y_{\texttt{Planet}}]$ with negligible bias. We can also see from Table 2, by applying the bias-variance decomposition, that the amount of irreducible error is very small.

Planet	$\mathrm{E}[y_{\texttt{Planet}} - f_{\texttt{Planet}}]$	$\mathrm{Var}[y_{\texttt{Planet}} - f_{\texttt{Planet}}]$	$\mathrm{E}[(y_{\texttt{Planet}} - f_{\texttt{Planet}})^2]$	$\mathrm{Var}[y_{\texttt{Planet}}]$
`Mercury`	1.20027e-03	8.76903e-04	8.78344e-04	2.96998e+04
`Venus`	-2.98130e-04	5.82933e-05	5.83822e-05	2.79233e+02
`Mars`	-5.46507e-05	9.26582e-05	9.26611e-05	2.50072e+03
`Jupiter`	8.16967e-05	1.67735e-04	1.67741e-04	5.78114e+02
`Saturn`	2.92417e-04	1.21947e-03	1.21955e-03	3.12429e+02

Table 2: The bias, variance, and mean squared error of the random forests, along with the variance of the $y_{\texttt{Planet}}$.

To see the overall shape of the empirical distribution of error, we make a Gaussian Kernel Density estimate for each set of residuals. Figure 2 demonstrates the sharp concentration of the residuals at 0 for all of the planets, demonstrating that the random forests indeed fit the targets well. Among all of the planets, the variance of the residuals for `Mercury` and `Saturn` are about an order of magnitude more than those for the other planets, indicating that the targets $y_{\texttt{Mercury}}$ and $y_{\texttt{Saturn}}$ were harder to learn. We see that this trend continues in our subsequent analysis.

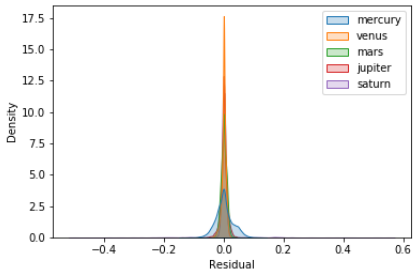

Figure 2: A Kernel Density Estimate of the Residual errors $y_{\texttt{Planet}} - f_{\texttt{Planet}}$.

We now apply Shapley Value Analysis to determine if the random forest correctly discarded the extraneous features and learned the inverse square relationship between $\|\ddot{\mathbf{r}}_{\texttt{Planet}}\|$ and $\|\mathbf{r}_{\texttt{Planet}}\|$. We

first verify that the extraneous features did not contaminate prediction. Figure 8 shows the SHAP summary plots for each of the planets. We can that for the planets `Mercury`, `Venus`, `Mars`, and `Jupiter`, the feature $\|\mathbf{r}_{\texttt{Planet}}\|$ was by far the most important, and the empirical distributions of ϕ_α, ϕ_δ, and ϕ_E are sharply concentrated about 0 as desired. For `Saturn`, however, we can see that ϕ_α is instead the most important feature, and and the empirical distributions of ϕ_δ and ϕ_E are less concentrated at 0, indicating that there was significant contamination from the extraneous features.

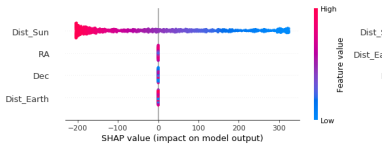

Figure 3: `Mercury`

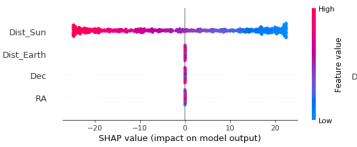

Figure 4: `Venus`

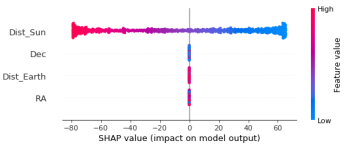

Figure 5: `Mars`

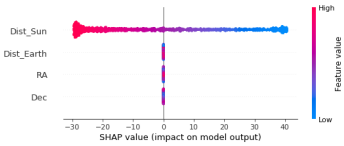

Figure 6: `Jupiter`

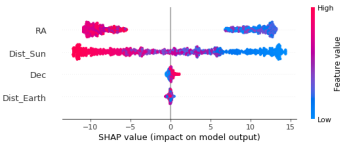

Figure 7: `Saturn`

Figure 8: SHAP Summary Plots for each of the planets.

The TSNE plots given in Figure 14 show the effect of contamination on the SHAP values. As stated in Section 1.2, without any contamination, these plots should trace a 1-dimensional manifold, as only one coordinate is nonzero. For the plots corresponding to `Venus`, `Mars`, and `Jupiter`, we can see this occurs exactly. The plot corresponding to `Mercury` is almost a contiguous 1-dimensional manifold, but we can see that the random forest $f_{\texttt{Mercury}}$ deals with the case of large values of $y_{\texttt{Mercury}}$ separately. The plot for `Saturn` clearly shows two distinct 1-dimensional manifolds, and we can further see from Figure 13 that one of the "branches" corresponds to values of $y_{\texttt{Saturn}}$ that are less than zero, and the other corresponds to values of $y_{\texttt{Saturn}}$ that are greater than zero.

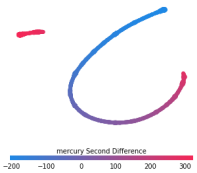

Figure 9: `Mercury`

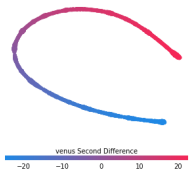

Figure 10: `Venus`

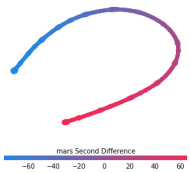

Figure 11: `Mars`

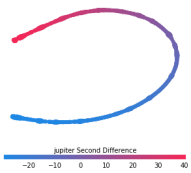

Figure 12: `Jupiter`

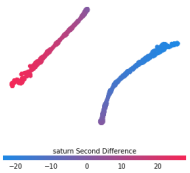

Figure 13: `Saturn`

Figure 14: TSNE Visualization of the SHAP values for each of the planets.

Finally, we can see that the random forests indeed learned the inverse square law from Figure 20. The dependence plots given in the top row show that for all planets except for `Saturn`, the function ϕ_S is continuous with respect to $\|\mathbf{r}_{\texttt{Planet}}\|$. In the middle row, we plot ϕ_S against the inverse square $\frac{1}{\|\mathbf{r}_{\texttt{Planet}}\|^2}$, and we can see that these plots are exactly linear with the exception of `Saturn`, which appears to be piecewise linear. In the bottom row, we visualize the true inverse square relationship present in the data by plotting $y_{\texttt{Planet}}$ against $\frac{1}{\|\mathbf{r}_{\texttt{Planet}}\|^2}$. This bottom row yields an explanation for why the random forest for `Saturn` had additional contamination: whereas the plots for the other planets are clean lines, the plot for `Saturn` has visible error from the inverse square relationship due to the limitations of the JPL Ephemeris system. This error is not pure noise, however, and has a period. The additional contamination can then be justified with this periodic noise correlating with the extraneous features that we added. Indeed, such correlation makes sense as the error is given by measurement error from Earth. Since these extra features pertain to Earth's position, it makes sense that these features correlate with the error, causing the random forests to split on the extraneous features.

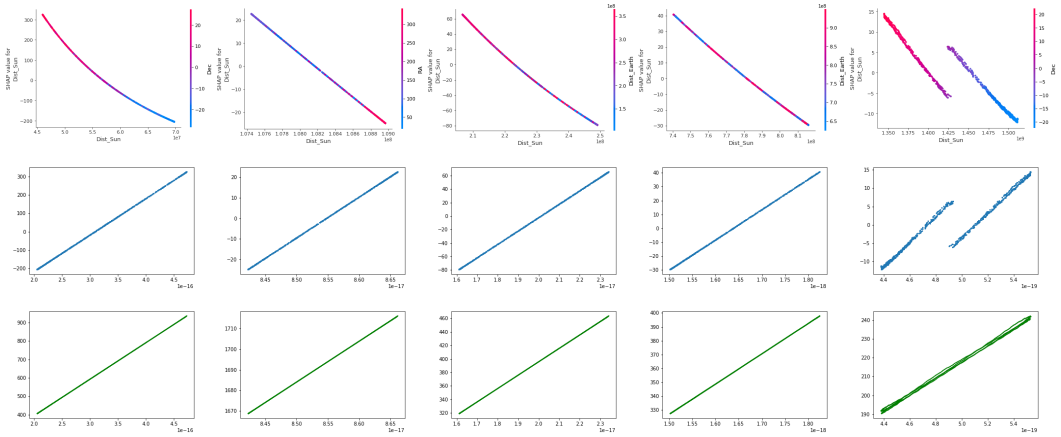

Figure 15: `Mercury` **Figure 16:** `Venus` **Figure 17:** `Mars` **Figure 18:** `Jupiter` **Figure 19:** `Saturn`

Figure 20: The top row consists of the SHAP dependence plots for each of the planets. The middle row is $\phi_S(\|\mathbf{r}_{\texttt{Planet}}\|)$ plotted against $\frac{1}{\|\mathbf{r}_{\texttt{Planet}}\|^2}$. The last row is $y_{\texttt{Planet}}$ plotted against $\frac{1}{\|\mathbf{r}_{\texttt{Planet}}\|^2}$.

4 Conclusion

Using tools from the explainability literature, we demonstrated that random forests, given data similar to that available to early physicists, are capable of identifying the feature that is causing a physical phenomenon, and describing the relationship between that feature and the observed phenomenon. Table 2 and Figure 2 show that random forests are well-suited for predicting the given targets. Figure 20 provides compelling evidence that given sufficiently accurate measurements, a random forest is able to implicitly infer the inverse square law as described in Section 1. Figures 8 and 14 demonstrate the ability of the random forest to prevent extraneous features, even those that have reason to be periodically correlated with the target, from contaminating the SHAP values of the causal feature. Finally, Figures 7, 13, and 19 show that random forests are capable of picking up hidden signals in the data that explain deviations in the observed trajectory of planets that are far away. With more precise data, we can extend our analysis to other celestial bodies, such as asteroids and planets far away from the sun.

We believe that this evidence demonstrates that the type of analysis that we carried out may be used to better understand more complicated causal relationships. A random forest can be trained on observational data, and the explanations ϕ_i given by Shapley Value analysis decompose a potentially complex signal into a linear combination of simple single-variate functions that may illuminate a certain causal relationship. In this way, a researcher may use this method to "test" the hypothesis that a given quantity has a succinct theoretical description in terms of some specified variables. A future line of research in this direction would be to apply this method to other problems in experimental physics to experimentally determine if certain quantities lacking a known analytical description may be approximately modeled by an analytical description generated by the Shapley Values of a random forest.

References

[APS+18] Astropy Collaboration, A. M. Price-Whelan, B. M. SipHocz, H. M. G"unther, P. L. Lim, S. M. Crawford, S. Conseil, D. L. Shupe, M. W. Craig, N. Dencheva, A. Ginsburg, J. T. Vand erPlas, L. D. Bradley, D. P'erez-Su'arez, M. de Val-Borro, T. L. Aldcroft, K. L. Cruz, T. P. Robitaille, E. J. Tollerud, C. Ardelean, T. Babej, Y. P. Bach, M. Bachetti, A. V. Bakanov, S. P. Bamford, G. Barentsen, P. Barmby, A. Baumbach, K. L. Berry, F. Biscani, M. Boquien, K. A. Bostroem, L. G. Bouma, G. B. Brammer, E. M. Bray, H. Breytenbach, H. Buddelmeijer, D. J. Burke, G. Calderone, J. L. Cano Rodr'iguez, M. Cara, J. V. M. Cardoso, S. Cheedella, Y. Copin, L. Corrales, D. Crichton, D. D'Avella, C. Deil, 'E. Depagne, J. P. Dietrich, A. Donath, M. Droettboom, N. Earl, T. Erben, S. Fabbro, L. A. Ferreira, T. Finethy, R. T. Fox, L. H. Garrison, S. L. J. Gibbons, D. A. Goldstein,

R. Gommers, J. P. Greco, P. Greenfield, A. M. Groener, F. Grollier, A. Hagen, P. Hirst, D. Homeier, A. J. Horton, G. Hosseinzadeh, L. Hu, J. S. Hunkeler, Z. Ivezi'c, A. Jain, T. Jenness, G. Kanarek, S. Kendrew, N. S. Kern, W. E. Kerzendorf, A. Khvalko, J. King, D. Kirkby, A. M. Kulkarni, A. Kumar, A. Lee, D. Lenz, S. P. Littlefair, Z. Ma, D. M. Macleod, M. Mastropietro, C. McCully, S. Montagnac, B. M. Morris, M. Mueller, S. J. Mumford, D. Muna, N. A. Murphy, S. Nelson, G. H. Nguyen, J. P. Ninan, M. N"othe, S. Ogaz, S. Oh, J. K. Parejko, N. Parley, S. Pascual, R. Patil, A. A. Patil, A. L. Plunkett, J. X. Prochaska, T. Rastogi, V. Reddy Janga, J. Sabater, P. Sakurikar, M. Seifert, L. E. Sherbert, H. Sherwood-Taylor, A. Y. Shih, J. Sick, M. T. Silbiger, S. Singanamalla, L. P. Singer, P. H. Sladen, K. A. Sooley, S. Sornarajah, O. Streicher, P. Teuben, S. W. Thomas, G. R. Tremblay, J. E. H. Turner, V. Terr'on, M. H. van Kerkwijk, A. de la Vega, L. L. Watkins, B. A. Weaver, J. B. Whitmore, J. Woillez, V. Zabalza, and Astropy Contributors. The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package. *aj*, 156(3):123, September 2018.

[ART+13] Astropy Collaboration, T. P. Robitaille, E. J. Tollerud, P. Greenfield, M. Droettboom, E. Bray, T. Aldcroft, M. Davis, A. Ginsburg, A. M. Price-Whelan, W. E. Kerzendorf, A. Conley, N. Crighton, K. Barbary, D. Muna, H. Ferguson, F. Grollier, M. M. Parikh, P. H. Nair, H. M. Unther, C. Deil, J. Woillez, S. Conseil, R. Kramer, J. E. H. Turner, L. Singer, R. Fox, B. A. Weaver, V. Zabalza, Z. I. Edwards, K. Azalee Bostroem, D. J. Burke, A. R. Casey, S. M. Crawford, N. Dencheva, J. Ely, T. Jenness, K. Labrie, P. L. Lim, F. Pierfederici, A. Pontzen, A. Ptak, B. Refsdal, M. Servillat, and O. Streicher. Astropy: A community Python package for astronomy. A&A, 558:A33, October 2013.

[Bre01] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[CEJB+07] D Richard Cutler, Thomas C Edwards Jr, Karen H Beard, Adele Cutler, Kyle T Hess, Jacob Gibson, and Joshua J Lawler. Random forests for classification in ecology. *Ecology*, 88(11):2783–2792, 2007.

[LEC+20] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, 2(1):2522–5839, 2020.

[LL17a] Scott M Lundberg and Su-In Lee. Consistent feature attribution for tree ensembles. *arXiv preprint arXiv:1706.06060*, 2017.

[LL17b] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*, pages 4765–4774, 2017.

[LLLW14] Xuqing Li, Xiangnan Liu, Meiling Liu, and Ling Wu. Random forest algorithm and regional applications of spectral inversion model for estimating canopy nitrogen concentration in rice. *J. Remote Sens*, 18(4):934–945, 2014.

[MH08] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[SBB+18] Emma VA Sylvester, Paul Bentzen, Ian R Bradbury, Marie Clément, Jon Pearce, John Horne, and Robert G Beiko. Applications of random forest feature selection for fine-scale genetic population assignment. *Evolutionary applications*, 11(2):153–165, 2018.

[Sha53] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.