Draft version March 22, 2023
Typeset using LaTeX default style in AASTeX631

Machine learning for the prediction of local asteroid damages

Gregoire Chomette,[1,2] Lorien Wheeler,[1] and Donovan Mathias[1]

[1] *NASA Ames Research Center, MS 258-5, Moffett Field, CA 94035, USA*
[2] *STC, 21 Enterprise Parkway, Suite 150, Hampton, VA 23666, USA*

ABSTRACT

Risk assessment studies of local asteroid hazards traditionally simulate the physics of meteors with engineering models tailored to analyze large numbers of scenarios. However, these simplified approaches still need to solve time-dependent ODEs to model the entry process and the resulting ground damage. With a computational cost of O(0.01 CPU.s) per scenario, the simulation of the tens-of-millions of possible entry conditions in risk assessment studies can take several days of CPU time without supercomputing resources. We propose in this paper an orthogonal approach based on machine learning models to predict the size of damaged areas given a list of entry parameters. We train 5 machine learning methods and compare the predictions to the the outputs of the PAIR model, and find that the neural network is the best model for the estimation of the radius of areas damaged by blast overpressure and thermal radiation, with around 10% average errors and a coefficient of determination (R2) of 0.99. The CPU time is decreased by a factor O(10^3) compared to the PAIR model, which enables the simulation of millions of scenarios in minutes, on a local computer. We then use the same machine learning approaches for a classification task where the models are trained to predict if an asteroid will produce a given level of damage. Results show that complex models like the gradient boosting classifier and the neural network can perform this task with 98% accuracy. Beyond surrogate models, we finally incorporate the machine learning algorithms to the state-of-the-art Shapley sensitivity analysis and present a ranking of the entry parameters based on their contributions to the ground damages.

Keywords: Machine learning, local asteroid damages, computational efficiency, sensitivity analysis

1. INTRODUCTION

Every day, Earth is bombarded with more than 100 tons of dust and sand-sized particles. While the majority of asteroids impacting the atmosphere burn up harmlessly before reaching the ground, dramatic evidence suggests that extreme consequences can result when large asteroids strike the Earth. For instance, the impact of extraterrestrial bodies was connected to global effects such as the historical K-T dinosaur extinction (Alvarez et al. (1980)), the Eltanin ocean impact (Gersonde et al. (1997)), and to local events like the impressive Meteor crater (Phillips et al. (1991)) or the flattened forest of Tugunska (Vasilyev (1998)). More recently, the unexpected airburst over Chelyabinsk led to over 1000 injured civilians and tens of millions of dollars to repair infrastructure damage (Popova et al. (2013)). Fast and robust predictive models are therefore required to assess asteroid threat levels and mitigate the hazards of future impacts.

Among the methods proposed to evaluate potential risks, high-fidelity simulations have been used to study specific impact events (Crawford et al. (1995), Boslough & Crawford (2008)), or to investigate particular physical mechanisms such as the thermal radiation effects of large asteroids (Johnston et al. (2018)), the influence of the height of burst on blast overpressure levels (Aftosmis et al. (2019)), or the effect of material models on energy deposition curves (Robertson & Mathias (2017)). As of today, high-fidelity numerical simulations are the most robust tools to simulate all the complex aerodynamic mechanisms arising during the entry of an asteroid in the Earth's atmosphere, but remain computationally very expensive, even with the recent gains of supercomputing power. To illustrate the latter point, the hydrocode simulations presented in the work of Robertson & Mathias (2017) required around 3500 core-hours to model each single case, making it impossible to use these models for the simulation of the tens-of-millions (Mathias et al. (2017)) of cases required for a statistical assessment of the risk levels.

Alternatively, a number of semi-analytical approaches have been published, employing a range of simplified models to tackle specific aspects of the problem. The trajectory of asteroids in the atmosphere are commonly simulated with the integration of single-body equations (Opik (2004)) tackled with ODE solvers. The wide variety of breakup behaviors can be modeled either with a continuous fragmentation (Hills & Goda (1993)) or a discrete fragmentation (ReVelle (2006)) approach. More recently, Wheeler et al. (2017, 2018) introduced the Fragment-Cloud Model (FCM), a hybrid method based on both the continuous and discrete fragmentation approaches with the ability to capture a broader range of physical processes. From the energy deposition curves computed by the FCM model, damage models have been developed to estimate the size of the areas damaged by blast overpressure with scaling relations (Hills & Goda (1993)) from nuclear sources (Glasstone et al. (1977)). Similarly, other damage models have put their focus on thermal hazards (Collins et al. (2005)), tsunamis, or global effects, providing tools to predict a broad range of asteroid damages without needing to run expensive, higher-fidelity CFD simulations.

In an effort to develop complete and fast-running risk assessment methods, the PAIR model (Mathias et al. (2017)) has been developed by NASA's Asteroid Threat Assessment Project (ATAP) to calculate several hazards of asteroids for a large number of realistic impacts. This approach combines state-of-the-art consequence simulation tools with a probabilistic framework where entry conditions are drawn from realistic statistical distributions. After demonstrating its usefulness in risk studies requiring the calculation of damage levels for tens-of-millions of impact cases, the PAIR model was a key capability supporting planetary defense preparedness, as demonstrated in hypothetical asteroid impact exercises. With the engineering models used in PAIR, the computational cost to simulate one asteroid impact scenario is O(0.01 CPU.seconds), an improvement of the time complexity of an order $O(10^{10})$ compared to some advanced, high-fidelity CFD simulations (Robertson & Mathias (2017)). However, running the tens-of-millions of asteroid impacts with the PAIR model still requires a couple of days of CPU time, which can quickly become prohibitive without distributed systems.

In an attempt to further improve the computational efficiency and integration of the risk models, another option is to capitalize on data-driven machine learning methods to predict asteroid risk levels. Indeed, machine learning approaches are known to be time efficient since models can, once trained, make predictions through simple function calls, decreasing the computational complexity to simulate scenarios by several orders of magnitude, compared to any numerical or semi-analytical technique. The proliferation of both datasets and machine learning libraries has removed the existing barriers of entry to develop data-driven surrogate models. In the field of planetary defense, several works have emerged to automate bolide (Smith et al. (2021)) or Near-Earth asteroid (Ye et al. (2019)) detection, infer meteoroid characteristics (Tárano et al. (2019)), or identify Earth-impacting asteroids (Hefele et al. (2020)) with neural networks. However, despite the availability of the data generated by PAIR (Mathias et al. (2017)), a machine learning model to predict the local damages of asteroids is still, to the best of our knowledge, lacking.

In this paper, we develop several machine learning models to predict the main local hazards of asteroids given a finite number of trajectory and material parameters. More specifically, we build five data-driven models with variable complexities to perform two assignments: (i) a classification task to identify scenarios yielding a specific damage level on the Earth's surface, and (ii) a regression task to predict the size of the damaged area. We train and test the machine learning models against synthetic datasets generated by the PAIR model (Mathias et al. (2017)), and benchmark the relative errors and coefficients of determination to assess the predictions. Additionally, we measure the inference time to predict damage levels with the machine learning models and compute the computational gain associated with these surrogate approaches. Then, we push the study with a state-of-the-art sensitivity analysis of our neural network models. To that end, we employ the Shapley method (Sundararajan & Najmi (2020)), tailored for machine learning models due to its high compuational cost, and get valuable information on how much each parameter contributes to increasing the local damages. We obtain a hierarchy among entry parameters (e.g., velocity, incidence angle, density, etc) in terms of their influence on the size of the damaged areas, and discuss how this analysis can help exploration teams know where to prioritize their focus.

2. METHODOLOGY

The data-driven approaches introduced in this paper belong to the class of supervised machine learning. The models are trained to make predictions on an output variable (i.e., the radius of a damaged area) based on some input variables (e.g., diameter of the asteroid, velocity, etc). The models are "supervised" by the dataset, considered as the ground truth, and try to learn functions that predict the output values as close as possible to the values present in the dataset. In this section, we first briefly describe how the synthetic dataset is generated, then provide an overview of the five machine learning models developed, and finally explain the procedure to train, tune and test the models.

2.1. *Data generation*

The dataset used to train the machine learning models contains a large number of realistic sub-300m asteroid impact scenarios causing local hazards. In order to generate each data point, we use the state-of-the-art PAIR model from Mathias et al. (2017) developed as part of NASA's Asteroid Threat Assessment Project (ATAP). This method can be broken down into two main steps: (i) the generation of fictitious entry conditions by sampling the entry parameters from realistic statistical distributions, and (ii) the calculation of local risk levels with semi-analytical models that capture the flight, break-up, blast and thermal processes. In the following we propose a brief overview of these two steps.

First, the generation of input parameters to characterize the atmospheric entry of an asteroid is based on a Monte Carlo framework where all the parameters are sampled independently from realistic statistical distributions. We select 8 asteroid parameters due to their ability to characterize the entry conditions: the diameter D, density ρ, aerodynamic strength S, velocity V, angle of incidence with respect to the horizontal θ, the strength-scaling exponent α informing the strength increase after fragmentation, the luminous efficiency η defining the fraction of energy emitted as thermal radiation, and the ablation σ informing the mass ablation rate. These parameters are sampled from the following distributions:

- The diameter (D) is a function of the H-magnitude (H) and the albedo (p_v) values such that $D = (1.326 \times 10^6) \times 10^{-H/5}/\sqrt{p_v}$, where H is sampled uniformly from 20 to 30, and p_v is sampled from the NEOWISE distribution (Mainzer et al. (2011)).
- The density (ρ) is highly dependant on the asteroid class and comes with great uncertainty, due to the measurement difficulties. Therefore, a Gaussian mixture model relying on several Gaussian meteorite densities (Macke (2010)) and fall fractions (Spohn et al. (2014)) is used, together with estimates of the overall asteroid macroporosity (Carry (2012)).
- The velocity (V) takes values in the [10;40] km/s range, with a distribution skewed towards 15-20 km/s (Greenstreet et al. (2012)).
- The entry angles of incidence (θ) are sampled from a cosine distribution such as $\theta \sim \frac{90}{\pi} cos(\frac{\pi}{90}\theta - \frac{\pi}{2})$, with $\theta \in [0; 90]$ degrees (Shoemaker (1962)). The incidence angles are computed with respect to the horizontal, at an altitude of 100 km.
- The macroscopic strength (S), luminous efficiency (η) and ablation coefficient (σ) are sampled from logarithmic distributions, respectively in the ranges [0.1; 10] MPa (Popova et al. (2011)), $[3 \times 10^{-4}; 3 \times 10^{-2}]$ (Mathias et al. (2017)), and $[3.5 \times 10^{-10}; 7 \times 10^{-8}]$ kg/J (Hills & Goda (1993); Collins et al. (2005); Shoemaker (1962)).
- The strength-scaling exponent (α), also defined in the following section, is usually sampled from a uniform distribution ranging from 0.1 to 0.3 (Mathias et al. (2017)).

After generating the list of parameters that fully characterize the asteroid entry conditions, the PAIR model simulates the behavior of the asteroid from atmospheric entry to ground impact, or when all the energy has been transferred to the atmosphere. The simulation starts at 100km above the surface of the Earth, and the flight dynamics of the asteroid are modeled by integrating single-body equations (Opik (2004)). Then, the break-up process is tackled with the fragment-cloud model (FCM) from Wheeler et al. (2017), combining traditional discrete and continuous fragmentation approaches. This method also tracks the associated energy deposited in the atmosphere and estimates the energy lost per unit altitude. The point of maximum energy deposition is taken to represent the approximate burst altitude of

the impact case, and the burst energy is assumed to be equal to the full initial kinetic energy of the asteroid. From the burst altitude and energy, the PAIR model then estimates the size of the area damaged by blast overpressure with a combination of scaling relations (Hills & Goda (1993)) from nuclear sources (Glasstone et al. (1977)), and height of burst maps obtained from CFD simulations (Aftosmis et al. (2019)). The thermal radiation damage is calculated based on the work of Collins et al. (2005) on ground-impact fireballs, with an extension to airbursts. For both the blast and thermal hazards, the PAIR model predicts the radius of circular areas damaged by blast overpressure or thermal radiation exceeding a given threshold. We use in this work the thresholds corresponding to four damage severities presented in Table 1.

Table 1. Levels of blast and thermal damages. The thresholds for overpressure Δp and total heating per unit area Φ are expressed in PSI and in MJ/m^2, respectively.

Damage level	Blast consequences	Thermal consequences
Serious	Windows break ($\Delta p = 1$)	2^{nd} degree burns ($\Phi = 0.25$)
Severe	Doors blow out ($\Delta p = 2$)	3^{rd} degree burns ($\Phi = 0.42$)
Critical	Most structures collapse ($\Delta p = 4$)	Clothing ignites ($\Phi = 0.84$)
Unsurvivable	Complete devastation ($\Delta p = 10$)	Sand explodes ($\Phi = 1.20$)

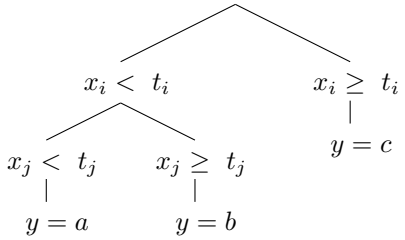
2.2. *Machine learning models overview*

The problem at hand consists of mapping a set of asteroid entry properties to the resulting local blast and thermal damages. From a mathematical standpoint, we are trying to find the best function $f : \mathbb{R}^n \rightarrow \mathbb{R}^+$ such that $f(X) = y$, where X is the vector containing all the asteroid entry conditions, and y is the radius of the circular area affected by a specific damage (e.g., $\Delta p = 1$ psi). More specifically, $X = [D, \rho, S, V, \theta, \alpha, \eta, \sigma]$ where D is the diameter, ρ is the density, S is the aerodynamic strength, V is the velocity, θ is the angle of incidence, α is the strength-scaling exponent, η is the luminous efficiency, σ is the ablation coefficient, and $y = r_{damaged}$ is the radius of the blast or thermally damaged region. In order to find the best function f, we propose to employ some of the latest machine learning models and to compare their accuracy, complexity, and interpretability. We implement 5 different models: a linear regressor, a decision tree, a random forest, a gradient booster, and a neural network. For completeness, we propose in the following a general overview of each model, assuming that $X = [x_1, x_2, ..., x_n]$ is a generic input vector containing the independent variables, and y is the dependent variable that we try to predict. More details on each model can be found in the work of Hastie et al. (2009).

The **linear regression** model is a statistical method used to model a linear relationship between the dependent variable and one or more independent variables. It is a parametric model with parameters β_0, β_1, ..., β_n that can be written as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n \tag{1}$$

A **decision tree** is a tree-like method which consists of making predictions after successively comparing the values of some independent variables x_i to some thresholds t_i. After going through all the branches of the tree, the model finds the best value for y, as illustrated in the following example:



The **random forest** model is made up of a large number of decision trees, where each decision tree is trained independently on different subsets of the training data. The final prediction of the random forest is made by averaging the predictions of all the individual trees, which traditionally helps to reduce overfitting, and makes the model more

robust. In mathematical terms, given n decision trees with predictions y_1, y_2, ..., y_n for an input vector X, the prediction of the random forest is:

$$y = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{2}$$

The **gradient booster** is also based on a combination of decision trees, but each tree is designed to correct the mistakes of the previous trees, rather than training the trees independently. The process starts by training a base decision tree on the training data, and by computing the residual errors between the predicted values and the observed values. Then, several iterations are performed where each new tree is trained on the previous residuals and added to the model. This process is repeated for a fixed number of trees, or until the performance of the model no longer improves. Mathematically, if n decision trees are created successively with y_0 being the prediction of the first decision tree, y_n being the prediction of the last one, and w_i being the weight of tree number i, the prediction of the gradient booster is:

$$y = \frac{1}{n}\sum_{i=1}^{n} w_i y_i \tag{3}$$

Finally, the **neural network** is a complex parametric function traditionally used to extract non-linear patterns in multidimensional datasets. In practice, a neural network is a succession of L layers where a layer l is characterized by a matrix W^l, a vector W_0^l containing a large number of weights, and an activation function f^l employed to introduce some degree of non-linearity. Each layer receives a vector X^{l-1} and transforms it into X^l such that:

$$X^l = f^l(W^l X^{l-1} + W_0^l) \tag{4}$$

where $X = X^0$ is the input vector containing the asteroid properties and $y = X^L$ is last layer computing the predicted output of the neural network. Using the symbol $W = \{W_0^1; W^1; W_0^2; W^2; ...; W_0^L; W^L\}$ to refer to all the weights of all the layers, the model can simply be written as:

$$y = f^L(W^L f^{L-1}(W^{L-1} f^{L-2}(... + W_0^{L-2}) + W_0^{L-1}) + W_0^L) = f(X, W) \tag{5}$$

2.3. *Training, tuning and testing of the models*

The first step in the development of machine learning models consists in preparing the data. In the present work, the 8 features of the dataset have different magnitudes, as evidenced by the aerodynamic strength (S) and the ablation coefficient (σ) taking values of the order 10^6 and 10^{-8}, respectively. In the interest of granting an equal importance to each entry parameter, we normalize the dataset with the SciPy MinMax scaler (Virtanen et al. (2020)) to transpose linearly each feature in the range $[0; 1]$. Then, we split the data into three independent subsets, namely the training, validation and test sets. The training set is used to train the models and obtain the model's weights, the validation set is used to tune the hyper-parameters, and the testing set is used to evaluate the final performance of the model on new, unseen data.

Once the data is normalized and split into the three subsets, we define a list of hyper-parameters that control some important architecture options. For instance, some important hyper-parameters are the maximum length of a decision tree, the number of trees in the random forest, the number of layers in the neural network, or the learning rate of an optimizer. For each of these hyper-parameters, we create a list of numerical values that we would like to investigate as we try to find the best functional. We create a plethora of models based on each combination of hyper-parameter values, and fit each model to the training set to obtain the optimal set of weights and thresholds. In order to decide which combination of hyper-parameters brings the best results, we define a metric to quantify the performance of the models, typically the coefficient of determination ($R2$) for a regression task, and the accuracy for a classification task. We evaluate this metric for each combination of hyper-parameters on the validation set, and select the option yielding the best results. This procedure is traditionally called a grid search on the hyper-parameters, and an example is provided in Appendix A for the tuning of the learning rate and regularizer, two hyper-parameters of the neural network model.

For each combination of hyper-parameters, training the machine learning models to obtain the optimal weights or thresholds is a complex process requiring advanced algorithms. Each machine learning method employs a different approach: the coefficients β_i of the linear regression model are computed with a gradient descent procedure (Ruder (2016)), while the models based on decision trees employ a top-down induction algorithm (Rokach & Maimon (2005))

to find the optimal thresholds t_i and corresponding independent variables x_i. Finally, the weights W of the neural network are trained with complex optimizers such as Adam (Kingma & Ba (2014)) or Adagrad (Duchi et al. (2011)).

The number of training, testing, and validation data points was set to 7000, 2000 and 1000, respectively. While larger datasets could have been generated with the PAIR model, training is a computationally expensive task with a cost increasing proportionally to the number of training points. We therefore limit our study to a dataset containing a total of 10^4 data points in order to properly explore the combinations of hyper-parameters for all the machine learning models. The data points belonging to the three subsets are drawn randomly from the main dataset, and we verify that they cover the whole range of entry conditions by plotting the distributions, as illustrated in Figure 1 for three entry parameters and each subset of data. We notice that the medians, quartiles and ranges remain similar among the subsets for each entry parameter. Moreover, the 10^4 points seem to cover the whole range of the distributions described previously. For instance, the diameter of the asteroids was sampled from the sub-300m range, which is confirmed by the whiskers of the three subsets in Figure 1 (left). The velocities of asteroids cover most of the $[10; 40]$ km/s range with a distribution skewed towards 15-20 km/s (middle), and the aerodynamic strength (S) box plots (right) also seem to be a good representation of the logarithmic sampling in the $[0.1; 10]$ MPa range defined in section 2.1.

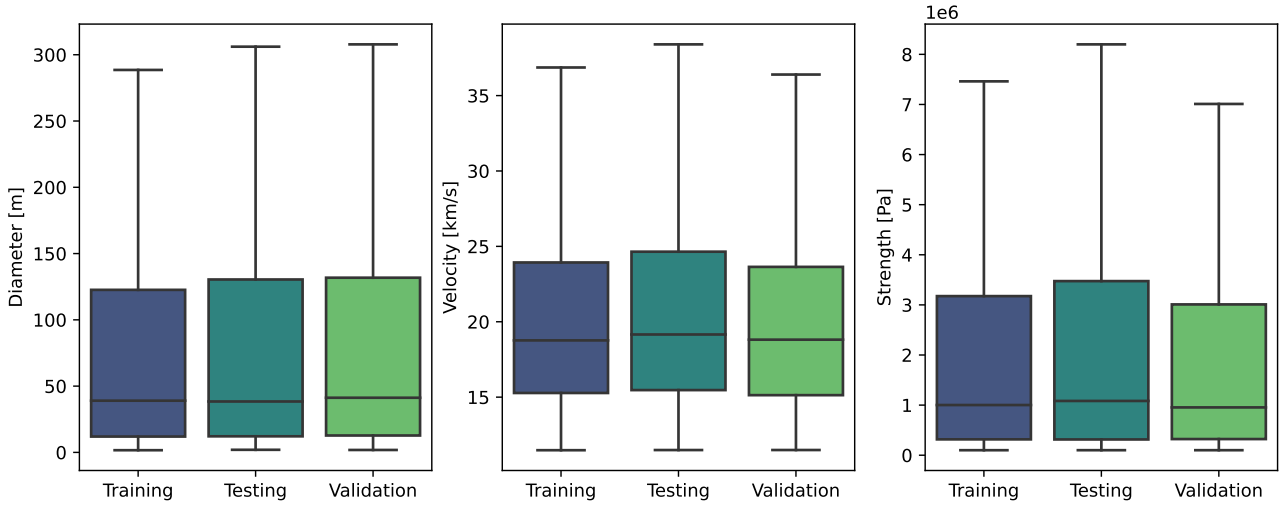


Figure 1. Box plots for the training, testing and validation sets, for three entry parameters. The box shows the median and quartiles of the dataset while the whiskers extend to show the rest of the distribution.

3. RESULTS

3.1. *Prediction of the size of damaged areas*

The five machine learning models described in the previous section were trained to predict the radius of areas seriously damaged by at least 1 PSI blast overpressure. The weights and thresholds of each model were optimized to minimize the squared errors between the predictions of the models and the values present in the dataset. A hyper-parameter tuning was performed to find the best settings of each model, and a summary of the settings retained is presented in Appendix B.

We show in Figure 2 for each machine learning model the predictions of serious blast damage, compared to the actual values of the test set. If the predictions were perfectly accurate, the data points would lie along a straight diagonal line from the bottom left to the top right of the plots. Some models, such as the neural network and to a lesser extent the random forest, gradient booster, and decision tree, produce a relatively narrow band of data points along this line. The linear regressor, on the other hand, makes highly inaccurate predictions except in the $50 - 100km$ radius range. We measure the goodness of the fit with the coefficients of determination ($R2$) which indicate the proportion

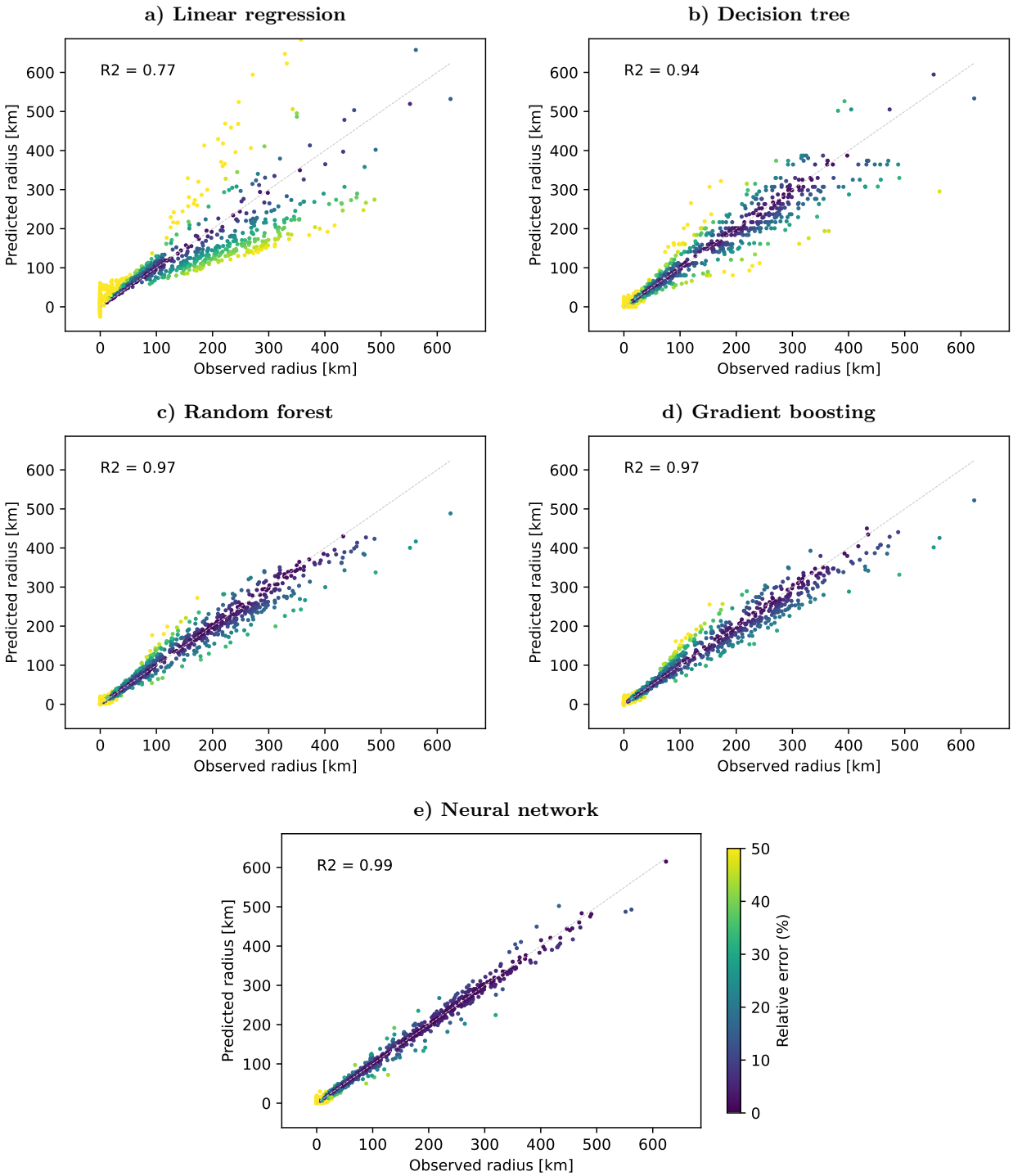


Figure 2. Prediction of the radii of the areas seriously damaged by $> 1PSI$ blast overpressure. The vertical axes feature the values predicted by the machine learning models, while the horizontal axes represent the values of the synthetic dataset, considered as the ground truth.

of the variance in the radii (the dependent variable) that can be predicted from the asteroid entry conditions (the independent variables). The coefficient of determination is defined as:

$$R2 = \frac{\sum_{i=1}^{N_{test}} (\hat{r_i} - r_i)^2}{\sum_{i=1}^{N_{test}} (\hat{r_i} - \bar{r})^2} \tag{6}$$

where $\hat{r}$ is the prediction, r is the ground truth, and i is the the index referring to a specific instance of the dataset. The $R2$ coefficients range from 0.77 to 0.99 for the five models, with higher values indicating better performances. Based on the coefficients of determination and initial visual inspection of the plots, the neural network appears to be the most accurate model among the ones considered. To gain further insights into the models, we measure the average absolute error of the predictions, defined as $\bar{e}_{abs} = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} |\hat{r_i} - r_i|$. We also compute the average relative error of the predictions $\bar{e}_{rel} = \frac{1}{N_{test}} \sum_{i=0}^{N_{test}} \frac{|\hat{r_i} - r_i|}{r_i}$, defined only for cases with a positive damaged area, i.e. where $r_i > 0$. In order to avoid extreme relative errors when the denominator approaches zero, we calculate the relative errors only for $r_i > 1km$. For scenarios where the damaged area is equal to zero (i.e., $r_i = 0$), we measure the average of the predictions of the machine learning models $\bar{e}_{abs}| = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \hat{r_i}$. These four metrics are selected to analyze and discuss the performance of our machine learning models. We present in Table 2 the results for 8 different neural network models trained to predict the radii of damaged areas, for the blast and thermal levels presented in Table 1.

Table 2. Summary of the performance of neural networks to predict the radii of damaged areas, for several blast and thermal damage levels. The metrics used to evaluate the accuracy are the coefficient of determination ($R2$), the average absolute errors ($\bar{e}_{abs}$), the average relative error for > 1 km damage scenarios ($\bar{e}_{rel}|_{r_i>1km}$), and the average predictions for 0 km damage scenarios ($\bar{e}_{abs}|_{r_i=0km}$).

Damage level	$R2$[%]	$\bar{e}_{abs}$ [km]	$\bar{e}_{rel}\|_{r_i>1km}$ [%]	$\bar{e}_{abs}\|_{r_i=0km}[km]$
Serious blast overpressure ($\Delta p = 1$ PSI)	0.99	5.0	11.5	2.0
Severe blast overpressure ($\Delta p = 2$ PSI)	0.99	2.4	11.3	1.6
Critical blast overpressure ($\Delta p = 4$ PSI)	0.99	1.7	9.7	0.7
Unsurvivable blast overpressure ($\Delta p = 10$ PSI)	0.99	0.8	8.1	0.4
Serious thermal radiation ($\Phi = 0.25$ MJ/m^2)	0.99	2.4	13.5	1.0
Severe thermal radiation ($\Phi = 0.42$ MJ/m^2)	0.99	2.2	14.1	1.0
Critical thermal radiation ($\Phi = 0.84$ MJ/m^2)	0.99	2.0	13.6	0.9
Unsurvivable thermal radiation ($\Phi = 1.20$ MJ/m^2)	0.99	1.5	12.2	0.9

The results in Table 2 show that the coefficients of determination are consistent across all tasks. Indeed, for all damage types and levels, 99% of the variance in the radii can be explained by the neural networks based on the 8 entry parameters. The average absolute errors of the predictions are in the range $[0.8; 5km]$ depending on the damage type considered. We know that the damaged areas are larger for milder damages for both the blast and thermal hazards, since the blast overpressure Δp and heating per unit area Φ decrease as we move away from the burst point. Thus, the differences in absolute errors presented in Table 2 show that the larger the radii is, the larger the absolute errors are. This remark is also confirmed by the observations on the predictions of the serious blast damage in Figure 3.1, where absolute errors tend to increase with larger observed radii. On the other hand, the relative errors $\bar{e}_{rel}|_{r_i>1km}$, tend to increase with smaller observed radii, as evidenced by Figure 3.1. The average relative errors are in the range $[8-14]$% across all the damage types considered for damaged areas with radii greater than 1km. Finally, for scenarios yielding no damage, the models tend to fail at predicting a zero damaged radius, with average predictions on the order of a few hundred meters to several kilometers. In these cases, the neural networks would in fact over-estimate the damages and predict that damage exists (i.e., false positive predictions) in more than 50% of the cases. The regression models presented in this section are therefore recommended for the prediction of sufficiently large asteroids, while smaller ones could be parsed differently.

3.2. *Identification of dangerous asteroids*

The neural networks discussed previously were found to be very effective in accurately predicting the size of damaged areas for a variety of damage types. However, it was noted that relative errors increase greatly for small radii, and that machine learning models are unable to distinguish between situations with small damage and no damage of a given level. In fact, neural networks are trained to minimize the sum of squared errors, and would not consider errors of a few kilometers to be significant, as the radii being predicted are typically on the scale of hundreds of kilometers. Therefore, the regression models of the previous section should be kept to predict the hazards of sufficiently large asteroids while smaller objects could be parsed differently according to their entry conditions to decide whether or not they require surveillance.

In this section we present the results of a binary classification method aimed at distinguishing between asteroids that generate a local damage of a given severity (e.g., a blast over-pressure greater than 1 psi), and those that do not. The output of this approach is a probability value, $p \in [0; 1]$, which indicates the likelihood that a given scenario belongs to the class of asteroids that produce damage of a given severity, as opposed to the class of asteroids that do not produce damage of this given severity. This classification approach modifies slightly the machine learning models described in section 2.2 to produce a probability $p \in [0; 1]$ instead of a radius $r \in \mathbb{R}^+$. In order to train the models, we add a new feature to the dataset consisting of a binary variable equal to 0 for scenarios yielding zero ground damage of a given severity, and equal to 1 for scenarios producing some damage of this severity. This categorical feature is the new dependent variable of interest, and the inputs of the models remain unchanged (i.e., $D, \rho, S, V, \theta, \alpha, \eta, \sigma$). The linear regressor is turned into a logistic regression model which outputs probabilities $p \in [0; 1]$ and is optimized to minimize the negative log likelihood loss. The decision tree regressor can be easily converted to a decision tree classifier by predicting class probabilities instead of radii values at each leaf node. The random forest and gradient booster are based on ensembles of decision trees, they can also predict probabilities with a weighted average of the probabilities of each decision tree. Finally, the last activation function of the neural network is set to a sigmoid function outputing values in the $[0; 1]$ range, and the weights are trained to minimize the binary cross-entropy with the new categorical variable. We train the models with the same number of data points as in the previous regression task, and present in Figure 3 an example of the results of the classification task for 1PSI blast overpressure radii, for two models.

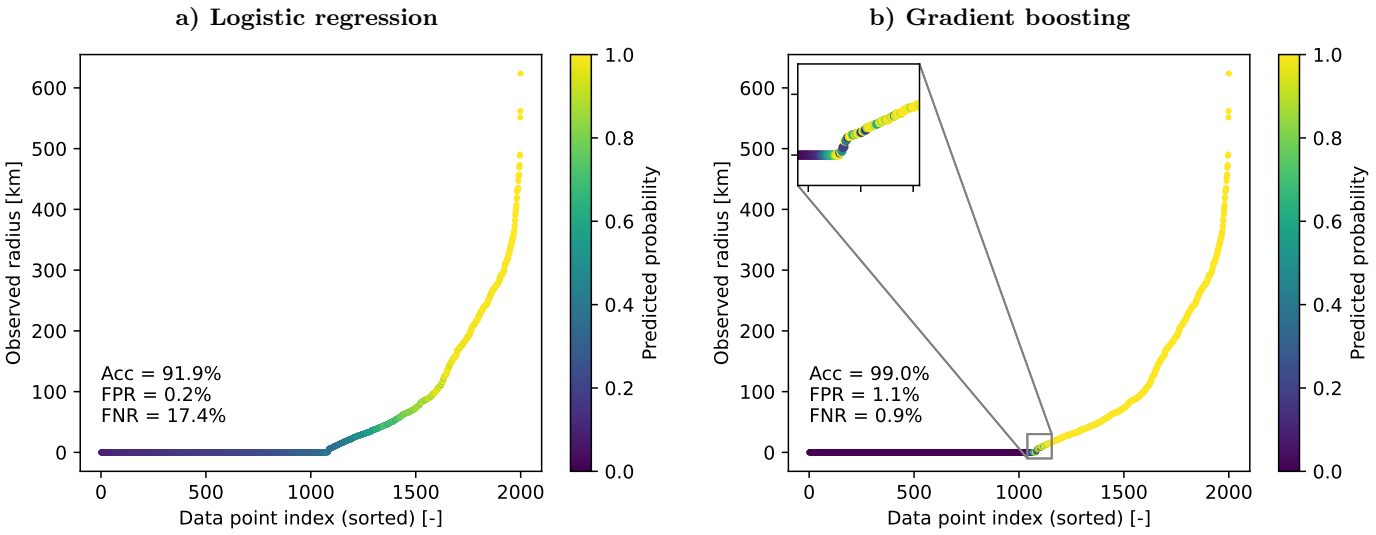


Figure 3. Results of the classification task for two machine learning models, the logistic regressor and the gradient booster. The 2000 data points of the test set are sorted horizontally according to their $1PSI$ blast radii predicted by the PAIR model, and the latter are indicated vertically. The color of the points correspond to the probabilities predicted by the classifiers, with large probabilities indicating that the asteroid is likely to produce at least $1PSI$ ground damage. Assuming that an asteroid produces a maximum blast over-pressure smaller and higher than $1PSI$ for $p < 0.5$ and $p \geq 0.5$ respectively, the accuracy, the false positive rate and the false negative rate are annotated on the plots.

The findings from the classification task depicted in Figure 3 indicate that most asteroids producing a maximum blast over-pressure smaller than $1PSI$ are given low probabilities of causing damage. On the other hand, models predict high damage probabilities for asteroids producing large areas damaged by $1PSI$ blast over-pressure. Both models show uncertainty in their predictions ($p \simeq 0.5$) mainly for small areas of damage, while events with damage radii greater than 20 and 100km are determined to be hazardous with a high level of certainty ($p \simeq 1$) for the gradient booster model and the logistic regressor, respectively. The gradient booster demonstrates greater confidence compared to the logistic regressor, with predicted probabilities tending towards the extremes, i.e. 0 and 1. We set the cut-off probability to $p = 0.5$ to identify asteroids producing $1PSI$ blast over-pressure on the ground, and notice that the gradient boosting model outperforms the logistic regression model, as evidenced by the overall classification accuracies of 99% and 91.9%. The logistic regression model particularly struggles in correctly identifying dangerous asteroids, with over 17% false negatives in this category.

The five classification machine learning models were trained and evaluated on test sets for the same classification task, and the results are detailed in Table 3. Except for the logistic regression model, all approaches exhibit accuracy levels above 97 % and a great balance between false positive and false negative rates. These accurate classification models are particularly interesting since they can complement the regression models that predict the radii of damaged areas. Indeed, the neural networks presented in the previous sections experienced difficulties in making predictions for small asteroids. The present classification methods are a potential solution to filter out asteroids producing damages smaller than a given threshold, and let the regression models estimate the size of the damaged area only for asteroids categorized as dangerous.

Table 3. Summary of the performance of the five machine learning models used for the classification task. The metrics used in this comparison are the accuracy, the false positive rate and the false negative rate, for a threshold probability $p = 0.5$.

Model	Accuracy (%)	False pos. rate (%)	False neg. rate (%)
Logistic regression	91.9	0.2	17.4
Decision tree	97.2	3.2	2.4
Random forest	98.2	1.3	2.3
Gradient boosting	99.0	1.0	1.3
Neural network	98.7	1.1	1.6

3.3. *Improvement of sensitivity analyses for model explanation*

In addition to the surrogate models presented above, machine learning techniques can bring new capabilities, for instance by enhancing the traditional sensitivity analyses employed to understand the importance of each entry parameter. Sensitivity studies play an important role to explain the comparative contribution of each feature (e.g., diameter, velocity, incidence angle, etc) to the local risk levels, generally by introducing small variations in each feature separately, and analyzing the influence on the output of the model. A more robust approach called the Shapley method (Gul (1989)) consists of computing the average predictions of a model for all possible input subsets, and measuring the change in model predictions when a feature is added to the set of input variables. The thorough evaluation of the model on all combinations of features adds accuracy to the analysis, but at a computational cost growing exponentially with the number of features and linearly with the number of data points. Thus, the Shapley analysis is computationally out of reach for traditional physics-based methods and tailored for cheaper machine learning models (Cohen et al. (2005); Jia et al. (2019)). In this section we apply the Shapley method to the neural networks developed previously to get better information on the contribution of each entry parameter to the size of the damaged area. A brief overview of the method is presented in the following.

The goal of the Shapley analysis is to understand the relative contributions of the inputs of a model in the determination of the output. In order to do so, the predictions of the model are computed for all possible feature subsets, with average values fed to the model when the features are removed. Each feature is then added, and the differences of the model's outputs with the feature and without it are tracked, normalized, and summed. The result of the summation

provides the Shapley value for a given data point and feature. The sign of the Shapley value indicates if the feature pushes the prediction of the model in a positive or negative direction, and its absolute value manifests the strength of the contribution to the output change. We present in Figure 4 the Shapley values for neural networks predicting serious blast (left) and thermal (right) damage, for 100 points from the test set.

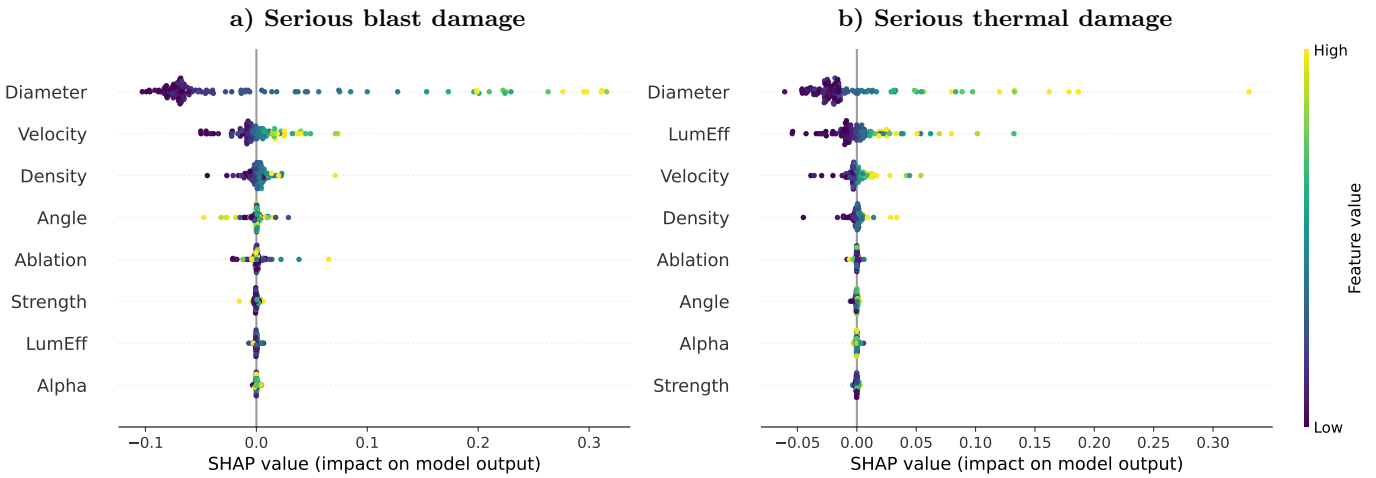


Figure 4. Shapley analyses for the prediction of serious blast and thermal damage areas from 8 entry condition features. The sign of the Shapley values indicate if the features increase or decrease the values of model output (i.e., the blast or thermal radius of the damaged area). The features are ranked vertically in ascending order based on their influence on the model output.

Figure 4 shows the contributions of the entry parameters to the size of damaged areas. The diameter, given the high absolute Shapley values associated with this feature, is found to be the most important entry parameter, and positively correlated to the model output with low and high feature values corresponding to low and high damage size, respectively. The velocity and density features also appear to play a significant role and to be positively correlated to the model output, while the contribution of the luminous efficiency is an important parameter only for thermal hazards. At the other end of the spectrum, the mechanical strength and the strength-scaling coefficient α, characterizing the evolution of the strength during the simulation, have a relatively low influence on risk levels.

4. DISCUSSION

The main motivation for developing new asteroid risk assessment tools based on data-driven methods is the potential computational efficiency gain. Conversely to traditional physics-based approaches, machine learning models infer the local hazards by simple and inexpensive function calls instead of solving systems of differential equations. Therefore, while simplified engineering models (Wheeler et al. (2017)) simulate each scenario with a computational cost of $O(0.01\ CPU.s)$ which results in several days of CPU time for the tens-of-millions of cases in risk assessment studies, machine learning models can cut the computational time by a factor $O(10^3)$ and perform these simulations in a few minutes. Alternatively, large risk assessment studies could be simulated quickly on local machines instead of supercomputers. Beyond the significant gain in computational efficiency, the latter models can also be easily integrated into risk assessment environments due to the democratization and standardization of machine learning libraries.

Among the possible approaches to replace physics-based methods, five machine learning models have been developed, trained and tested on separate datasets. Results revealed that they are capable of predicting the size of damaged areas with coefficients of determination ranging from 0.77 to 0.99 depending on the complexity of the models. More specifically, the linear model presents the largest errors, while more advanced methods estimate local risk levels with great accuracy. Therefore, results suggest that the relationship between entry condition and local damages is highly non-linear, and that risk levels can't be predicted accurately with simple and interpretable models. In particular, the neural network is the most suited for this task with average relative errors around 10% for most of the damage range, and 99% of the variance predictable from independent variables. The capacity of neural networks to extract complex patterns and correlations between the different asteroid entry conditions makes it the optimal model among those considered. The binary classification task identifying asteroids producing a given damage level exhibits similar

results: missclassification errors drop by a factor $6-8$ from a simple logistic regression model to the more advanced gradient boosting and neural network classifiers for most damage types. The latter models can classify asteroids with an accuracy around 98%, and are particularly useful in the low damage range, precisely where the relative errors of regression models are significant. In fact, classification and regression models can be used complementarily, the first one being employed to decide whether the scenario is dangerous, and the second one estimating the size of the damaged area, for scenarios identified as dangerous. For both tasks, we would recommend non-linear models for their better accuracy, despite their relatively low interpretability. Based on the results of both tasks, the neural network appears to be the most interesting model for local risk assessment, followed by the gradient boosting and random forest models.

Machine learning models can be further refined to improve their alignment with the damage levels projected by physics-based approaches, if necessary. Possible strategies include augmenting the synthetic dataset with additional training points or features to characterize the entry of asteroids. Alternatively, classification models can be tuned by picking the optimal threshold probability to identify scenarios producing a given level of damage. Decreasing the threshold value p allows, for instance, to reduce drastically the number of false negative scenarios that could have been wrongly neglected. However, questions need to be raised about putting significant efforts on fitting machine learning models very precisely to a dataset generated through engineering models and not immune to uncertainties. To the best of our knowledge, the entry, break-up, and hazard propagation models employed to generate the dataset with PAIR (Mathias et al. (2017)) do not provide uncertainty quantification metrics, and could produce errors in the estimation of damaged areas in comparison to higher fidelity simulations. In the absence of such data reliability metrics, 98% accuracy in the classification task and 10% average relative errors in the prediction of the blast and thermal radii can be considered satisfactory.

In addition to providing accurate and cost-effective surrogate models, machine learning techniques offer capabilities that are beyond the scope of traditional physics-based methods. The Shapley method discussed in the preceding section exemplifies this by conducting a sensitivity analysis that enables the prioritization of input parameters based on their impact on the extent of damage. The method calculates the average damage size across all possible feature subsets and for a substantial number of data points, making it computationally infeasible for any physics-based model. In contrast, machine learning models can execute a full Shapley analysis with a low time complexity, thereby providing a better explanation of the model's predictions and surpassing most existing model explanation approaches. The Shapley analysis yields several key findings, such as the diameter, velocity, and density being the most influential factors in determining the damage size, ranked in decreasing order for both blast and thermal damages. Additionally, the analysis reveals the luminous efficiency, a highly uncertain parameter in the thermal model, to have a significant contribution to the damage size. This finding highlights the need for alternative models that rely less on this parameter to accurately assess thermal risk levels. Conversely, the aerodynamic strength and strength-scaling coefficients are found to have negligible effects on the risk level assessment models for both blast and thermal hazards. These insights can guide the development of new models that are less dependent on uncertain parameters, such as the luminous efficiency, or identify the physical quantities that require closer attention in observation missions.

5. CONCLUSION

This work presents new data-driven approaches to assess the risk levels of large numbers of asteroid impact scenarios. The proposed machine learning methods reduce the CPU time required to model the entry process and resulting ground damage by $O(10^3)$ in comparison to the state-of-the-art engineering approaches, thus enabling large risk studies to be performed on local portable machines, in a few minutes. The machine learning models are trained and tested with synthetic datasets generated with the PAIR model, and are capable of estimating the radius of a damaged area with around 10% relative error. Other classification approaches are introduced to predict if a small impactor produces local damages greater than a given threshold, and these models categorize correctly the scenarios 98% of the time. Moreover, we capitalize on the computational efficiency of machine learning models to present a state-of-the-art sensitivity analysis of the size of the damaged areas with respect to each entry parameter. We hierarchize the entry conditions as a function of their contribution to the blast and thermal hazards. Results allow us to formulate recommendations on where to prioritize efforts regarding the characterization of hazardous asteroids, and raise concern about including the very uncertain and influent luminous efficiency parameter in thermal models.

APPENDIX

A. EXAMPLE OF HEATMAP FOR HYPER-PARAMETER TUNING

Figure 5 below illustrates the grid search procedure followed for hyper-parameter optimization. In this example, several values for the learning rate and regularizer are tested on a validation dataset, and the coefficient of determination (R2) is computed. We finally select one of the combinations of hyper-parameters yielding the highest R2.

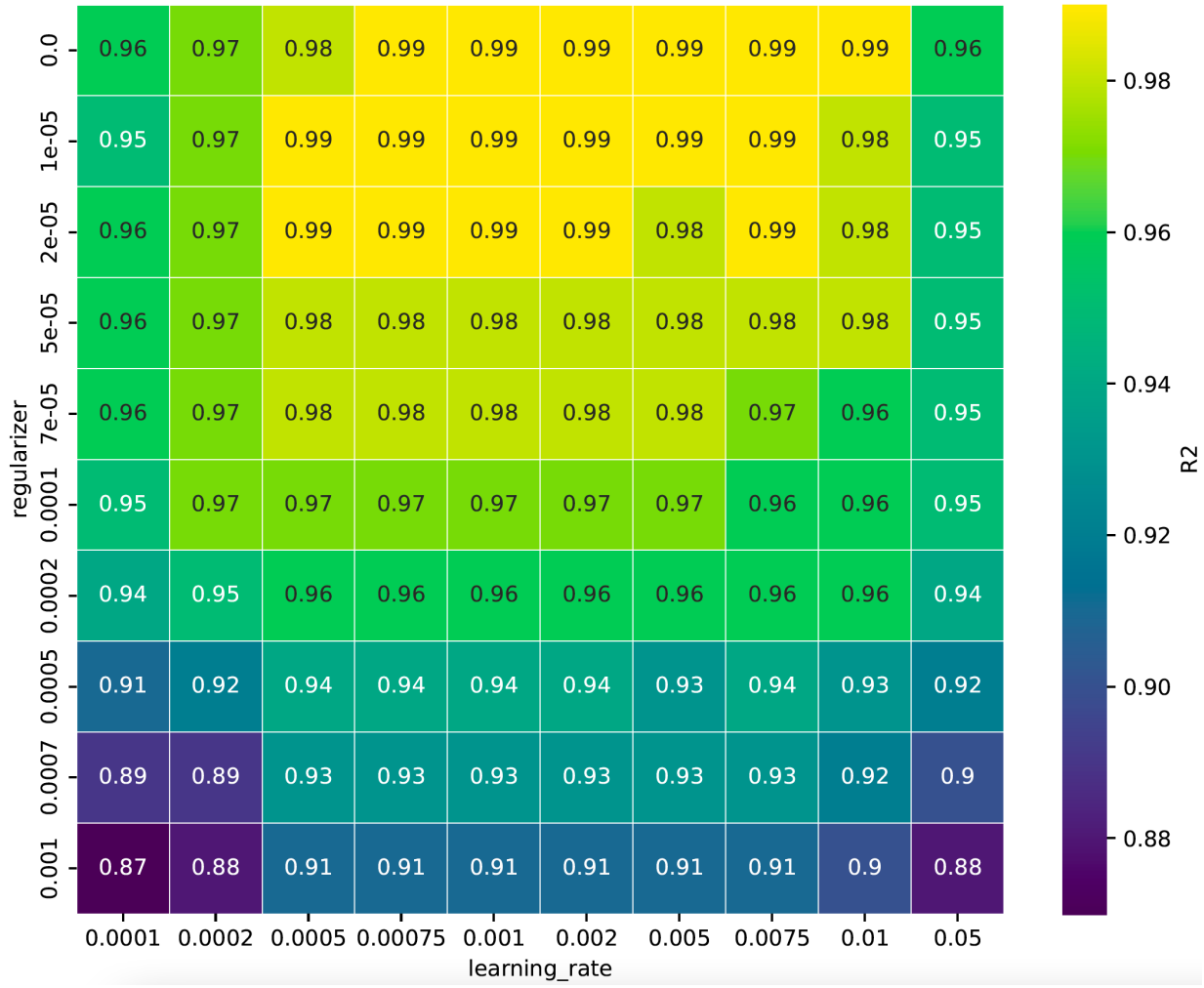


Figure 5. Heatmap of the hyper-parameter optimization, for the neural network in the regression task

B. SUMMARY OF THE SELECTION OF HYPER-PARAMETERS

The following Table 4 presents the final sets of hyper-parameters selected for each model, after fine-tuning on a validation set. If other hyper-parameters are not specified, default values from the TensorFlow (Abadi et al. (2015)) and Keras (Chollet et al. (2015)) libraries are retained for the neural network, and default values from SciPy (Virtanen et al. (2020)) are used for the remaining models.

Table 4. Summary of the hyper-parameters retained for each model, for the regression and classification tasks

Model	Hyperparameters (regression)	Hyperparameters (classification)
Decision tree	Maximum depth: 8	Maximum depth: 3
Random forest	Maximum depth: 8	Maximum depth: 12
Gradient boosting	Maximum depth: 4 Subsample: 0.8 Learning rate: 0.05	Maximum depth: 4 Subsample: 0.8 Learning rate: 0.5
Neural network	Learning rate: 0.005 Regularizer: 0.0 Batchsize: 64 Epochs: 50 Number of layers: 3 Number of units: [32,64,128] Hidden activation functions: 'relu' Output activation function: 'linear' Early stopping epoch number: 10	Learning rate: 0.01 Regularizer: 0.0 Batchsize: 64 Epochs: 50 Number of layers: 3 Number of units: [32,64,128] Hidden activation functions: 'sigmoid' Output activation function: 'linear' Early stopping epoch number: 10

REFERENCES

Abadi, M., Agarwal, A., Barham, P., et al. 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/

Aftosmis, M. J., Mathias, D. L., & Tarano, A. M. 2019, Acta Astronautica, 156, 278

Alvarez, L. W., Alvarez, W., Asaro, F., & Michel, H. V. 1980, Science, 208, 1095

Boslough, M., & Crawford, D. 2008, International Journal of Impact Engineering, 35, 1441

Carry, B. 2012, Planetary and Space Science, 73, 98

Chollet, F., et al. 2015, Keras, https://keras.io

Cohen, S., Ruppin, E., & Dror, G. 2005, other words, 1, 155

Collins, G. S., Melosh, H. J., & Marcus, R. A. 2005, Meteoritics & planetary science, 40, 817

Crawford, D., Boslough, M., Trucano, T., & Robinson, A. 1995, International journal of impact engineering, 17, 253

Duchi, J., Hazan, E., & Singer, Y. 2011, Journal of machine learning research, 12

Gersonde, R., Kyte, F., Bleil, U., et al. 1997, Nature, 390, 357

Glasstone, S., Dolan, P. J., et al. 1977, 50

Greenstreet, S., Ngo, H., & Gladman, B. 2012, Icarus, 217, 355

Gul, F. 1989, Econometrica: Journal of the Econometric Society, 81

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. 2009, The elements of statistical learning: data mining, inference, and prediction, Vol. 2 (Springer)

Hefele, J. D., Bortolussi, F., & Zwart, S. P. 2020, Astronomy & Astrophysics, 634, A45

Hills, J. G., & Goda, M. P. 1993, The Astronomical Journal, 105, 1114

Jia, R., Dao, D., Wang, B., et al. 2019, in The 22nd International Conference on Artificial Intelligence and Statistics, PMLR, 1167–1176

Johnston, C. O., Stern, E. C., & Wheeler, L. F. 2018, Icarus, 309, 25

Kingma, D., & Ba, J. 2014, International Conference on Learning Representations

Macke, R. J. 2010

Mainzer, A., Grav, T., Bauer, J., et al. 2011, The Astrophysical Journal, 743, 156

Mathias, D. L., Wheeler, L. F., & Dotson, J. L. 2017, Icarus, 289, 106

Opik, E. J. 2004, Physics of meteor flight in the atmosphere (Courier Corporation)

Phillips, F. M., Zreda, M. G., Smith, S. S., et al. 1991, Geochimica et Cosmochimica Acta, 55, 2695

Popova, O., Borovička, J., Hartmann, W. K., et al. 2011, Meteoritics & Planetary Science, 46, 1525

Popova, O. P., Jenniskens, P., Emel'yanenko, V., et al. 2013, Science, 342, 1069

ReVelle, D. O. 2006, Proceedings of the International Astronomical Union, 2, 95

Robertson, D., & Mathias, D. 2017, Journal of Geophysical Research: Planets, 122, 599

Rokach, L., & Maimon, O. 2005, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 35, 476

Ruder, S. 2016, arXiv preprint arXiv:1609.04747

Shoemaker, E. M. 1962, Physics and Astronomy of the Moon, 283

Smith, J. C., Morris, R. L., Rumpf, C., et al. 2021, Icarus, 114576

Spohn, T., Breuer, D., & Johnson, T. 2014, Encyclopedia of the solar system (Elsevier)

Sundararajan, M., & Najmi, A. 2020, in International conference on machine learning, PMLR, 9269–9278

Tárano, A. M., Wheeler, L. F., Close, S., & Mathias, D. L. 2019, Icarus, 329, 270

Vasilyev, N. V. 1998, Planetary and Space Science, 46, 129

Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, Nature Methods, 17, 261, doi: 10.1038/s41592-019-0686-2

Wheeler, L. F., Mathias, D. L., Stokan, E., & Brown, P. G. 2018, Icarus, 315, 79

Wheeler, L. F., Register, P. J., & Mathias, D. L. 2017, Icarus, 295, 149

Ye, Q., Masci, F. J., Lin, H. W., et al. 2019, Publications of the Astronomical Society of the Pacific, 131, 078002