

Научная статья

УДК 531.39, 52-17

doi: 10.17223/19988621/76/7

Применение методов машинного обучения для классификации резонансного движения астероидов

Татьяна Юрьевна Галушина¹, Елизавета Александровна Николаева²,
Дмитрий Сергеевич Красавин³, Оксана Никитична Летнер⁴

^{1, 3, 4} Томский государственный университет, Томск, Россия

² Тартуский университет, Тарту, Эстония

¹ tanastra@nxt.ru

² volna@sibmail.com

³ iosfixed@gmail.com

⁴ oksana.letner@gmail.com

Аннотация. При изучении динамики резонансных астероидов возникает необходимость классификации критических аргументов в зависимости от поведения на циркуляцию, либрацию и смешанный случай. Для автоматизации этой задачи предлагается использовать методы машинного обучения, такие как искусственные нейронные сети и HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise). В результате экспериментов по подбору параметров получена обученная модель, способная классифицировать резонансное движение.

Ключевые слова: астероид, орбитальный резонанс, машинное обучение, искусственная нейронная сеть

Благодарности: Исследование выполнено за счет гранта Российского научного фонда (проект № 19-72-10022).

Для цитирования: Галушина Т.Ю., Николаева Е.А., Красавин Д.С., Летнер О.Н. Применение методов машинного обучения для классификации резонансного движения астероидов // Вестник Томского государственного университета. Математика и механика. 2022. № 76. С. 87–100. doi: 10.17223/19988621/76/7

Application of machine learning methods for the classification of asteroid resonance motion

Tatyana Yu. Galushina¹, Elizaveta A. Nikolaeva²,
Dmitriy S. Krasavin³, Oksana N. Lenter⁴

^{1, 3, 4} Tomsk State University, Tomsk, Russian Federation

² University of Tartu, Tartu, Estonia

¹ tanastra@nxt.ru

² volna@sibmail.com

³ iosfixed@gmail.com

⁴ oksana.letner@gmail.com

Abstract. When studying the resonant asteroid dynamics, it is necessary to classify time series of critical arguments on circulation, libration, or mixed case depending on their behavior. It is logical to use modern methods of machine learning to automatize this process. Earlier, a similar problem was solved for artificial satellites of the Earth.

The purpose of this paper is to adapt the software attended for distinguishing resonant and nonresonant motion of satellites to solving asteroid dynamics problems. To achieve this goal, it is necessary to modify the program code and to train the created model on time rows obtained during the study of the asteroid orbital evolution.

Operation of the modified software can be divided into three stages. At the first stage, to simplify the model-classifier, we make coding of time series of asteroid resonant arguments by vectors of lower dimension using an artificial neural network – an autoencoder. The second stage includes automatic clustering time series of asteroid resonant arguments by the HDBSCAN method (Hierarchical Density-Based Spatial Clustering of Applications with Noise) and their manual labeling to learn the classifier. At the third stage, based on the obtained training set, the artificial neural network-classifier is learned.

The results of the classifier operation are estimated by visual comparison of graphs of the time series and received assessments. We may conclude that the classifier works correctly in most cases; some inaccuracies are observed in case of extreme amplitude and in the mixed case when libration passes to circulation.

Keywords: asteroid, mean motion resonance, machine learning, artificial neural network

Acknowledgments: This work was supported by the Russian Science Foundation (Scientific Project № 19-72-10022).

For citation: Galushina, T.Yu., Nikolaeva, E.A., Krasavin, D.S., Lenter, O.N. (2022) Application of machine learning methods for the classification of asteroid resonance motion. *Vestnik Tomskogo gosudarstvennogo universiteta. Matematika i mekhanika – Tomsk State University Journal of Mathematics and Mechanics*. 76. pp. 87–100. doi: 10.17223/19988621/76/7

Введение

В настоящее время динамика астероидов, сближающихся с Землей (АСЗ), привлекает все большее внимание ученых, что связано с рядом причин, одна из которых – их опасность для нашей планеты [1]. На сегодняшний день выявление потенциально опасных для Земли астероидов и исследование эволюции их орбит

является важной и актуальной задачей, поскольку на крайне высоких скоростях падение даже небольшого небесного тела может привести как к региональной, так и к глобальной катастрофе.

Необходимо также отметить, что орбитальные резонансы играют существенную роль в изучении движения рассматриваемых объектов. Способствуя поддержанию геометрической конфигурации «астероид–планета», устойчивый резонанс является своего рода механизмом защиты астероида от сближения с соответствующей планетой, в то время как неустойчивый резонанс приводит к регулярным сближениям и хаотичности движения. Таким образом, неустойчивые орбитальные резонансы могут вызывать тесные и многократные сближения АСЗ с большими планетами, что, в свою очередь, приводит к существенным изменениям параметров орбит астероидов и увеличивает вероятность их столкновений с планетами [2].

При исследовании большого числа реальных или виртуальных астероидов возникает необходимость визуально оценить значительное количество графиков с целью классификации движения на резонансное, нерезонансное или смешанное. Использование ручного труда не только занимает много времени, но и может приводить к ошибкам вследствие невнимательности, свойственной людям при выполнении монотонной работы. Естественным является желание привлечь современные методы машинного обучения для автоматизации данного процесса. Ранее подобная задача была решена для искусственных спутников Земли [3, 4].

Цель настоящей работы – адаптировать программно-алгоритмическое обеспечение, предназначенное для разделения резонансного и нерезонансного движения спутников, к решению задач астероидной динамики. Для достижения указанной цели требуется модифицировать программный код и обучить созданную модель на временных рядах, полученных в ходе исследования орбитальной эволюции астероидов.

Резонансные характеристики

Явление орбитального резонанса возникает вследствие соизмеримости периодов (или средних движений) астероида и большой планеты. Это значит, что если отношение периодов астероида и планеты можно представить как отношение двух взаимно простых целых чисел, то эти небесные тела движутся в орбитальном резонансе. В таком случае возмущения, вызванные взаимным гравитационным влиянием тел друг на друга, имеют периодический характер и усиливаются в момент соединения астероида и планеты. В зависимости от определенных условий резонансы могут оказывать как стабилизирующее воздействие на орбиту, тем самым защищая астероиды от сближений и столкновений с планетами, так и дестабилизирующее, что значительно изменяет орбиту и приводит к хаотичности движения [2].

Резонансный (критический) аргумент β и его производная по времени α , называемая резонансной щелью [5], являются основными характеристиками резонансного движения. Следует отметить, что при нахождении α производными аргумента перицентра ω_0 и долготы восходящего узла Ω_0 пренебрегают, так как изменения этих величин несут существенные по сравнению с изменениями средних долгот астероида и планеты. Усиление возмущений происходит в момент соеди-

нения астероида и планеты, откуда следует, что выражение для долготы соединения является основным [2]. Тогда резонансный аргумент определяется как

$$\beta = k_1\lambda_1 - k_2\lambda_2 - (k_1 - k_2)\omega_1 - (k_1 - k_2)\Omega_1, \quad (1)$$

а резонансная щель имеет вид:

$$\alpha \approx k_1n_1 - k_2n_2. \quad (2)$$

где n_1, n_2 – средние движения, λ_1, λ_2 – средние долготы астероида и планеты соответственно, ω_1 – аргумент перигентра астероида, Ω_1 – долгота восходящего узла астероида, k_1, k_2 – целые положительные числа. Порядок резонанса определяется величиной $k = k_1 + k_2$.

В соответствии с [2], если α и β испытывают колебания около значения точной соизмеримости, так что $|\beta - \beta_{\text{ср}}| \leq 180^\circ$ и $|\alpha| \leq \alpha_{\text{max}}$, то астероид движется в окрестности резонанса. Здесь $\beta_{\text{ср}}$ – центр либрации резонансного аргумента, а величина α_{max} характеризует границы резонансного движения и определяется по максимальной амплитуде колебаний резонансного аргумента β .

Таким образом, либрация резонансного аргумента говорит о наличии устойчивого резонанса, в то время как при его циркуляции резонанс отсутствует. Существует также смешанный случай, когда либрация сменяется циркуляцией или наоборот, что указывает на наличие неустойчивого резонанса.

Используемые методы

Машинное обучение – это подход к анализу данных, который включает в себя построение и адаптацию математических моделей, позволяющих программам «учиться» на опыте. Алгоритмы машинного обучения можно разделить на четыре основных категории: обучение с учителем, обучение с частичным привлечением учителя, обучение без учителя и обучение с подкреплением [6, 7]. В данной работе используются методы обучения с учителем и без учителя, поэтому далее представим описание только для них.

Задачу одного из основных видов машинного обучения, а именно задачу обучения с учителем, можно описать следующим образом [8]. Пусть имеется множество объектов, которые описываются парой переменных – признаковой и целевой. Пусть X – это множество допустимых признаковых переменных, а Y – множество допустимых целевых переменных. Пусть также имеется некоторая целевая функция $f : X \rightarrow Y$. Вид этой функции, вообще говоря, неизвестен, она не обязательно является математической функцией, но известны ее значения $y_i = f(x_i)$ на некотором подмножестве $\{x_1, \dots, x_t\} \subset X$. При этом пары объектов (x_i, y_i) формируют множество $T = \{(x_1, y_1), \dots, (x_t, y_t)\}$, называемое обучающей выборкой. Задача машинного обучения заключается в том, чтобы подобрать некоторую параметрическую функцию $p : X \rightarrow Y$, которая наилучшим образом приближала бы целевую функцию f на всем множестве T , а также ее параметры.

При обучении без учителя обучающая выборка содержит только множество признаковых объектов $T' = \{x_1, \dots, x_t\}$. Целью алгоритма обучения без учителя является создание модели, которая принимает на вход объект x_i и затем преобра-

зует его в некоторое значение или вектор в зависимости от типа рассматриваемой проблемы.

В данной работе решаются задачи кодирования временных рядов векторами низкой размерности, кластеризации и классификации кодов временных рядов резонансных аргументов астероидов. Для этого применяются такие методы машинного обучения, как искусственные нейронные сети (классификация), автокодировщики (уменьшение размерности), *k*-means и HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) (кластеризация).

Искусственная нейронная сеть (ИНС) – это математическая модель, построенная по принципу организации и функционирования биологических нейронных сетей, а также ее программное воплощение. Структура ИНС представляет собой совокупность узловых слоев, содержащую входной слой, один или несколько скрытых слоев и выходной слой. Через входной слой внешние данные поступают в сеть, в скрытых слоях выполняется основная обработка данных, а выходной слой выполняет финальные вычисления и несет в себе результат работы сети.

Как и любой алгоритм машинного обучения, искусственная нейронная сеть обучается на наборе данных. В зависимости от вида обучения (с учителем, с частичным привлечением учителя, без учителя, с подкреплением) процесс обучения ИНС будет происходить по-разному, однако в качестве основных его принципов можно выделить обработку данных нейронами, обновление значений весов в соответствии с определенным правилом и повторение всех шагов обучающего алгоритма до тех пор, пока не будет минимизирована некоторая функция потерь.

Автокодировщик – это искусственная нейронная сеть, которая обучается восстанавливать входной сигнал на выходе. Такую ИНС можно рассматривать как сеть, состоящую из двух частей: кодировщика и декодировщика. Искусственные нейронные сети с данной архитектурой имеют симметричную по количеству нейронов в слоях структуру относительно среднего слоя. Этот слой описывает так называемый код (или же скрытое представление), являющийся отображением исходного сигнала в пространство меньшей размерности.

Процесс обучения автокодировщика заключается в следующем: на вход сети подается набор признаков X , кодировщик переводит этот входной набор в его скрытое представление $Z = f(X)$, а декодировщик затем производит реконструкцию сигнала, получая на выходе набор $X' = f(Z)$. Таким образом, автокодировщик стремится получить выходные значения, близкие к целевым, т.е. к X , минимизируя функцию потерь. В данной работе функция потерь представляет собой логарифм гиперболического косинуса ошибки прогнозирования и имеет следующий вид:

$$L = \sum_i \log(\cosh(g_i)), \quad (3)$$

где $g_i = \sum_j |x'_i - x_i|$, x'_i – выходное значение, x_i – целевое, $i = \overline{1..t}$, $j = \overline{1..d}$, t – размер обучающей выборки, а d – размерность признаковой переменной.

Метод *k*-means, или же метод *k*-средних, является алгоритмом машинного обучения без учителя, используемым для кластеризации объектов. Цель данного алгоритма – разбить множество объектов $T' = \{x_1, \dots, x_m\}$ на k кластеров таким образом, чтобы каждый объект в своем кластере походил на другие объекты

из этого же кластера больше, чем на объекты из какого-либо другого кластера. Метод k -средних использует итеративный подход для получения конечного результата.

HDBSCAN – это основанный на плотности расположения объектов алгоритм иерархической пространственной кластеризации приложений с шумом. Основанные на плотности алгоритмы кластеризации объединяют объекты в кластеры путем поиска таких областей в пространстве данных, которые имеют высокую плотность и при этом окружены областями с меньшей плотностью. Особенность HDBSCAN как алгоритма иерархической кластеризации состоит в том, что он способен идентифицировать кластеры с переменной плотностью благодаря созданию иерархии вложенных кластеров.

Описание программно-алгоритмического обеспечения

Язык программирования Python наилучшим образом подходит для написания программ в сфере машинного обучения вследствие наличия большого числа библиотек и больших возможностей для визуализации данных и результатов. Работу модифицированной программы можно разделить на три этапа. На первом этапе для упрощения модели классификатора и ускорения его обучения проводится кодирование временных рядов резонансных аргументов астероидов векторами более низкой размерности. Второй этап заключается в автоматической кластеризации временных рядов резонансных аргументов астероидов и их последующей ручной разметке для обучения классификатора. На основе полученной обучающей выборки на третьем этапе происходит обучение классификатора.

Перед выполнением вышеперечисленных задач входные данные, которые представляют собой текстовые файлы с временными рядами, разбиваются на обучающую и тестовую выборки в процентном соотношении 80/20. Это осуществляется запуском отдельной программы, которая каждый файл случайным образом определяет в одну из выборок. Обучающая выборка предназначена для построения модели, в то время как тестовая – для проверки построенной модели.

Решение первой задачи осуществляется путем обучения автокодировщика. Для этого запускается основной скрипт с набором параметров, необходимых для задания конфигурации программы, который, в свою очередь, вызывает скрипт с алгоритмом автокодировщика. После завершения обучения и тестирования получаем модель, которая позволяет преобразовывать временные ряды, уменьшая их размерность до заданного значения. Далее осуществляется запуск программы, используемой для получения скрытых представлений временных рядов и записи их в файл.

Для решения второй задачи используется отдельная программа, которая выполняет кластеризацию скрытых представлений временных рядов методами k -means и HDBSCAN, а также их разметку. Для разметки вручную указываются номера кластеров, относящихся к либрации (метка 1), циркуляции (метка 0) или смешанному типу движения (метка 0.5), после чего каждый временной ряд сопоставляется с соответствующей ему меткой с помощью цикла. В результате получается набор данных, состоящий из размеченных скрытых представлений временных рядов.

Третья задача решается с помощью программы, реализующей искусственную нейронную сеть, способную классифицировать временные ряды. На данном эта-

не используются методы обучения с учителем, где обучающей выборкой служат данные, полученные на предыдущем шаге. Обученная ИНС позволяет сопоставить временной ряд уменьшенной размерности с числом в диапазоне от 0 до 1, которое определяет близость объекта к одному из трех классов.

Результаты и обсуждение

Исходные данные представляют собой 27 432 текстовых файла, содержащих временные ряды критических (резонансных) аргументов астероидов в радианах. Каждый временной ряд состоит из 1 024 элементов, что существенно усложняет работу с данными, поэтому выполняется несколько экспериментов по их кодированию векторами меньшей размерности, направленных на подбор оптимального размера скрытого представления. Чем больше размер скрытого представления, тем более сложные временные ряды удастся представлять с его помощью. В качестве начального приближения берется любое число, а затем в зависимости от полученных результатов необходимо либо повышать размерность, либо понижать. Если результат представления временных рядов получается неудовлетворительным, то выполняется повышение размерности. Если же результат получается хорошим или даже отличным, то осуществляется понижение размерности, так как всегда есть вероятность, что выбранная размерность скрытого представления является избыточной.

Результаты работы автокодировщика просматриваются с помощью графиков, изображающих исходные и восстановленные ряды для обучающей и тестовой выборок. Эти графики отбираются случайным образом и отражают разное поведение критического аргумента. Кроме того, строятся графики функции потерь L (3) для тестовой выборки в зависимости от номера эпохи. Эпоха в машинном обучении представляет собой одно полное прохождение обучающего набора данных через алгоритм.

Первый эксперимент проводился с размерностью скрытого представления, равной 24. Пример графиков исходного и восстановленного рядов для обучающей и тестовой выборок, полученных в этом случае, представлен на рис. 1 (a – обучающая, b – тестовая), а график функции потерь изображен на рис. 2. На рис. 1 исходный ряд отмечен серым цветом, а восстановленный – черным.

Из рис. 1 видно, что при данном размере скрытого представления автокодировщик очень хорошо восстановил временной ряд, о чем к тому же свидетельствует низкое значение ошибки. По окончании данного эксперимента сформировался набор весов модели, который принимался за основу при обучении моделей в последующих экспериментах. За счет этого уменьшилось количество эпох, что позволило сократить время обучения с четырех дней до одного.

Далее эксперимент был повторен для значений размерности скрытого представления 16, 12 и 8. Значения функции потерь на последней эпохе для всех экспериментов по уменьшению размерности вынесены в табл. 1, что позволяет более наглядно представить ошибку в каждом эксперименте. После сравнения всех результатов становится очевидным, что наилучшая точность достигалась при размере скрытого представления, равном 24, поэтому модель, полученная при таком значении, использовалась при последующей работе с временными рядами.

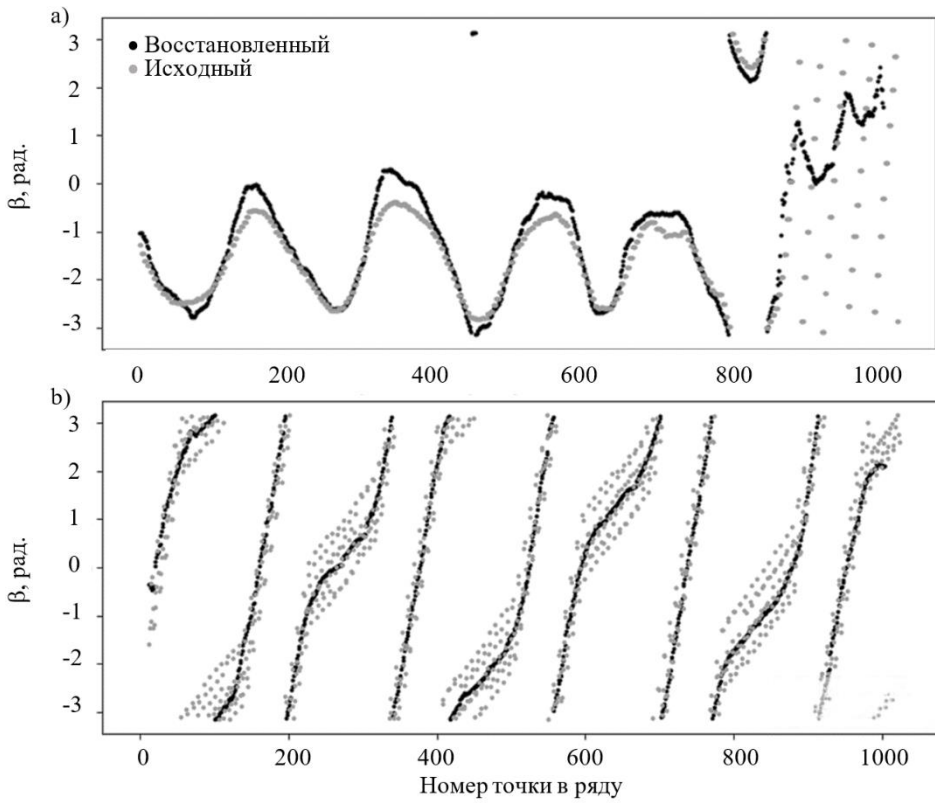


Рис. 1. Исходный (серый) и восстановленный (черный) ряды для (a) обучающей выборки и (b) тестовой выборки (размер скрытого представления равен 24)

Fig. 1. Original (gray) and reconstructed (black) series for (a) training and (b) test sets (the dimension of the hidden representation is 24)

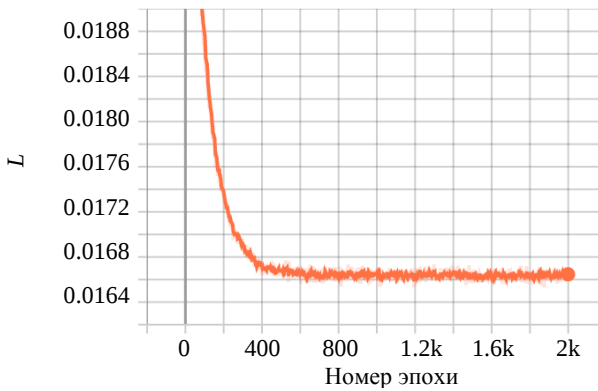


Рис. 2. График функции потерь для тестовой выборки (размер скрытого представления равен 24)

Fig. 2. Graph of the loss function for the test set (the dimension of the hidden representation is 24)

Таблица 1

Значения функции потерь на последней эпохе

Размеры скрытого представления	Значения функции потерь
8	0.02515
12	0.02273
16	0.02271
24	0.01663

Первый эксперимент по кластеризации проводился с использованием метода *k*-means, основным параметром которого является количество кластеров *k*. В ходе эксперимента данный метод вызывался три раза для различных значений *k*, равных 20, 25 и 30. Остальные параметры метода оставались постоянными при всех запусках, чтобы имелась возможность провести сравнение между результатами.

Использование метода *k*-means для кластеризации данных временных рядов не принесло достаточно хороших результатов. Во всех трех случаях самые большие кластеры содержат не единообразные объекты, т.е. объекты, не относящиеся к одинаковому типу движения. Это говорит о том, что многие объекты определены алгоритмом в неподходящие кластеры. Подобные результаты прослеживаются и во многих меньших кластерах, однако кластеры с единообразными объектами все же присутствуют. Следует отметить, что для применения данного метода необходимо хотя бы примерно понимать, какое количество кластеров должно образоваться. Это делает его неудобным и слабо применимым к данным текущей задачи, так как в ней невозможно предугадать точное количество кластеров.

Для следующего эксперимента использовался более мощный метод кластеризации HDBSCAN, важной особенностью которого является то, что он учитывает шум в данных путем объединения объектов, не отнесенных ни к одному кластеру, в кластер с меткой -1 . При необходимости объекты из кластера с такой меткой можно пропустить через метод повторно. Параметрами метода [9], которые требуют экспериментального подбора, являются следующие величины:

- 1) `min_cluster_size` (минимальный размер кластера): определяет минимальное число объектов, которое алгоритм рассматривает как кластер;
- 2) `min_samples` (минимальное количество образцов): определяет, какое количество образцов должно находиться в окрестности объекта, чтобы он рассматривался как основной. Чем больше это число, тем больше точек принимается за шум;
- 3) `cluster_selection_epsilon` (величина ϵ): выступает как некоторое граничное значение для разделения кластеров на меньшие группы. Например, если $\epsilon = 0.5$, то алгоритм не разделяет кластеры, расстояние между которыми меньше этого значения.

Описанные параметры оказывают влияние друг на друга и имеют несколько неочевидный эффект, поэтому необходимо многократно запустить алгоритм HDBSCAN для выявления их наиболее оптимального сочетания. В данной работе алгоритм запускался 7 раз для различных значений `min_cluster_size`, `min_samples` и `cluster_selection_epsilon`, которые подбирались опытным путем с опорой на их смысл и выдаваемый результат. Эти значения приведены в табл. 2 вместе с соответствующим им количеством кластеров, образовавшихся по завершении работы HDBSCAN.

**Значения параметров HDBSCAN и соответствующее им количество
получившихся кластеров**

Номер эксперимента	min_cluster_size	min_samples	cluster_selection_epsilon	Количество кластеров
1	20	1	0.5	58
2	50	5	0.5	39
3	100	1	0.5	29
4	100	1	0.7	25
5	150	50	0.3	23
6	200	15	0.5	24
7	200	15	0.7	23

Оценка результатов кластеризации осуществляется визуально с помощью графиков временных рядов, отобранных случайным образом из каждого кластера. Наименее пригодные результаты получились в эксперименте 7, так как в двух из самых больших кластеров оказались объекты, не относящиеся к одному типу движения. Наиболее удачным оказался эксперимент 3, по завершении которого как самые крупные, так и более мелкие кластеры содержали объекты с единым типом движения, хотя некоторая неточность в одном небольшом кластере все же присутствовала. Примеры объектов, отнесенных в данном эксперименте к одному кластеру, показаны на рис. 3, где каждый столбец представляет собой отдельный кластер. Из рис. 3 видно, что кластер 11 соответствует смешанному типу движения, 9 – либрации, 4 – циркуляции.

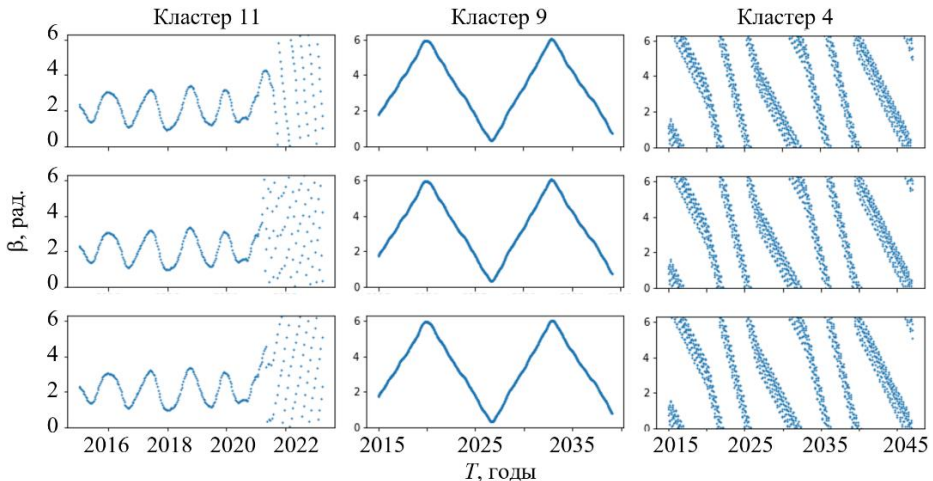


Рис. 3. Примеры объектов в кластерах 11, 9 и 4, получившихся в эксперименте 3

Fig. 3. Examples of objects in clusters 11, 9, and 4 obtained in experiment 3

Отметим, что в данном эксперименте менее 10% всех данных алгоритм определил в кластер –1. В связи с небольшим размером этого кластера было принято

решение не проводить его повторную кластеризацию и, соответственно, не включать объекты этого кластера в обучающую выборку для ИНС-классификатора, а затем использовать их в качестве тестовых объектов.

Кластеры, полученные в эксперименте 3 (см. табл. 2), использовались далее для разметки объектов в соответствии с типом движения, что является необходимым шагом перед обучением ИНС-классификатора. Каждый временной ряд сопоставляется с меткой 0 (циркуляция), 1 (либрация) или 0.5 (смешанный тип) в соответствии с технологией, описанной выше. Результаты обученной ИНС-классификатора на данный момент можно оценить только зрительным способом с помощью графиков, где изображен временной ряд критического аргумента астероида. На каждом графике подписано число в диапазоне от 0 до 1, определяющее близость движения к либрации, циркуляции или смешанному типу. Это число также можно интерпретировать как вероятность того, что конкретный временной ряд соответствует устойчивому резонансу.

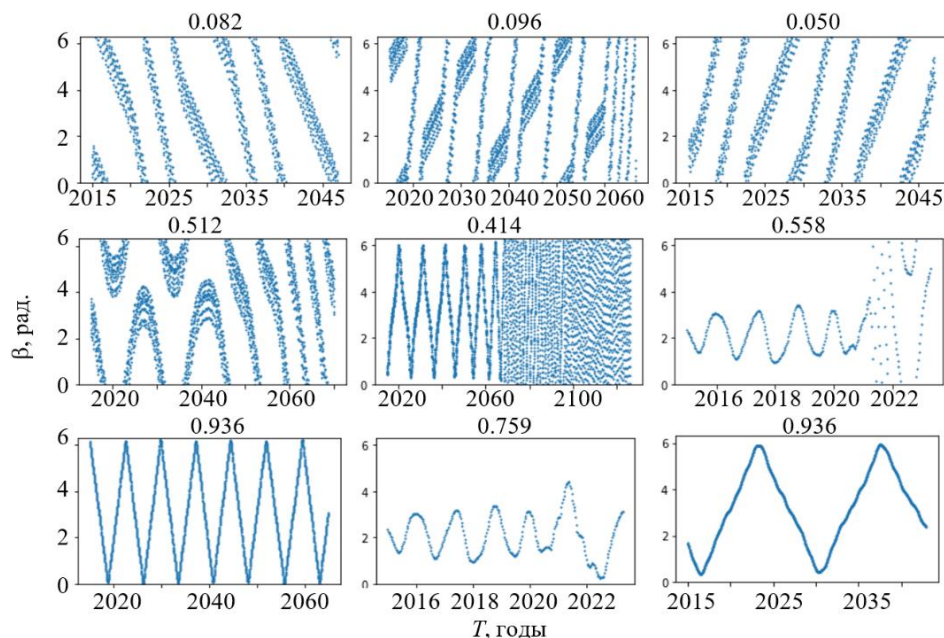


Рис. 4. Примеры объектов, оцененных ИНС в процессе классификации
Fig. 4. Examples of objects evaluated by an artificial neural network during the classification process

На рис. 4 представлены некоторые примеры объектов, оцененных ИНС в процессе классификации, где видно, что присвоенные объектам значения соответствуют типам их движения. В процессе визуальной оценки результатов определено, что большинству объектов классификатор присвоил корректные метки, однако в некоторых случаях присутствуют явные неточности. Примеры неудачной классификации приведены на рис. 5, где можно увидеть, что ИНС дает ошибочную оценку типа движения в случаях, когда амплитуда колебаний критического аргумента является предельной.

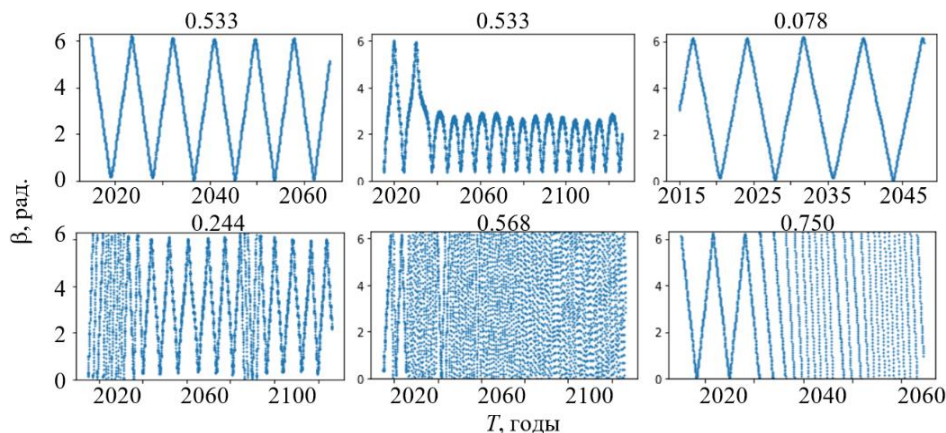


Рис. 5. Примеры неудачной классификации объектов

Fig. 5. Examples of unsuccessful classification of objects

Заключение

В данной работе выполнена адаптация программно-алгоритмического обеспечения, предназначенного для разделения резонансного и нерезонансного движения спутников, к решению задач астероидной динамики, полученная модель обучена на выборке временных рядов резонансных аргументов астероидов. На первом этапе с целью выбора оптимального размера скрытого представления проведен эксперимент с разными значениями, лучшие результаты показала размерность 24. На следующем этапе кластеризации протестированы методы k -means и HDBSCAN с различными значениями параметров. HDBSCAN показал лучшие результаты, которые были использованы для ручной разметки кластеров. На третьем этапе на основе полученной разметки проводилось обучение ИНС-классификатора, результатом работы которого является число от 0 до 1, которое можно интерпретировать как вероятность наличия устойчивого резонанса. Результат работы классификатора оценивается визуально сопоставлением графиков временных рядов и полученных оценок. Можно сделать вывод, что в большинстве случаев классификатор работает корректно, некоторые неточности наблюдаются в случае либрации с предельной амплитудой и в смешанном случае, когда предельная либрация переходит в циркуляцию.

Список источников

1. Астероидно-кометная опасность: вчера, сегодня, завтра / под ред. Б.М. Шустова, Л.В. Рыхловой. М. : Физматлит, 2010. 384 с.
2. Галушина Т.Ю. Орбитальные и вековые резонансы в движении астероидов, сближающихся с Землей // Физика космоса : тр. 49-й Междунар. студенческой науч. конф., Екатеринбург, 27–31 января 2020 г. Екатеринбург : УрФУ, 2020. С. 6–18.
3. Красавин Д.С., Александрова А.Г., Томилова И.В. Применение искусственных нейронных сетей в задачах анализа динамической структуры областей околоземного орбитального пространства // Известия вузов. Физика. 2020. Т. 63, № 3. С. 70–75.

4. Красавин Д.С., Александрова А.Г., Томилова И.В. Применение искусственных нейронных сетей в исследовании динамической структуры околоземного орбитального пространства // Известия вузов. Физика. 2021. Т. 64, №10. С. 38–43.
5. Гребеников Е.А., Рябов Ю.А. Резонансы и малые знаменатели в небесной механике. М. : Наука, 1978. 128 с.
6. Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. 2nd ed. Sebastopol : O'Reilly Media, 2019. 484 p.
7. Burkov A. The Hundred-Page Machine Learning Book. 2019. 152 p. URL: <http://ema.cri-info.cm/wp-content/uploads/2019/07/2019BurkovTheHundred-pageMachineLearning.pdf> (accessed: 27.07.2021).
8. Nilsson N.J. Introduction to machine learning. URL: <http://robotics.stanford.edu/people/nilsson/MLBOOK.pdf> (accessed: 26.04.2021).
9. Parameter Selection for HDBSCAN : HDBSCAN Clustering Library Documentation. URL: https://hdbscan.readthedocs.io/en/latest/parameter_selection.html (accessed: 09.01.2021).

References

1. Asteroidno-kometnaya opasnost: vchera, segodnya, zavtra [Asteroid-comet hazard: yesterday, today, tomorrow] (2010) Ed. by Shustova B.M., Rykhlovoy L.V. Moscow: Fizmatlit.
2. Galushina T.Yu. (2020) Orbitalnyye i vekovyye rezonansy v dvizhenii asteroidov, sblizhayushchikhsya s Zemley [Mean motion and secular resonances in the motion of near-Earth asteroids]. *Proceeding of the 49-th International Student Conference "Physics of space", Ekaterinburg, 2020*. pp. 6–18. DOI: 10.15826/B978-5-7996-2935-9.01.
3. Krasavin D.S., Aleksandrova A.G., Tomilova I.V. (2020) Application of artificial neural networks to an analysis of the dynamic structure of the near-Earth orbital space. *Russian Physics Journal*. 63(3). pp. 426–431. DOI: 10.1007/s11182-020-02053-z.
4. Krasavin D.S., Aleksandrova A.G., Tomilova I.V. (2022) Application of artificial neural networks in studying the dynamic structure of the near-Earth orbital space. *Russian Physics Journal*. 64(10). pp. 1824–1830.
5. Grebenikov E.A., Ryabov Y.A. (1978) *Rezonansy i малыe znamenateli v nebesnoy mekhanike* [Resonances and small denominators in celestial mechanics] Moscow: Nauka.
6. Géron A. (2019) *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2nd edition*. Sebastopol: O'Reilly Media.
7. Burkov A. (2019) *The Hundred-Page Machine Learning Book*. Andriy Burkov.
8. Nilsson N.J. (2021) *Introduction to machine learning*. Access mode: <http://robotics.stanford.edu/people/nilsson/MLBOOK.pdf>
9. (2021) *Parameter Selection for HDBSCAN* [Electronic resource]: HDBSCAN Clustering Library Documentation. Access mode: https://hdbscan.readthedocs.io/en/latest/parameter_selection.html

Сведения об авторах:

Галушина Татьяна Юрьевна – кандидат физико-математических наук, заведующая лабораторией компьютерного моделирования и машинного анализа астрономических данных НИИ прикладной математики и механики Томского государственного университета (Томск, Россия). E-mail: tanastra@nxt.ru

Николаева Елизавета Александровна – магистрант Тартуского университета (Тарту, Эстония). E-mail: volna@sibmail.com

Красавин Дмитрий Сергеевич – младший научный сотрудник лаборатории компьютерного моделирования и машинного анализа астрономических данных НИИ прикладной

математики и механики Томского государственного университета (Томск, Россия). E-mail: iosfixed@gmail.com

Летнер Оксана Никитична – кандидат физико-математических наук, старший научный сотрудник лаборатории компьютерного моделирования и машинного анализа астрономических данных НИИ прикладной математики и механики Томского государственного университета (Томск, Россия). E-mail: oksana.letner@gmail.com

Вклад авторов: все авторы сделали эквивалентный вклад в подготовку публикации. Авторы заявляют об отсутствии конфликта интересов.

Information about the authors:

Galushina Tatyana Yu. (Candidate of Physics and Mathematics, Tomsk State University, Tomsk, Russian Federation). E-mail: tanastra@nxt.ru

Nikolaeva Elizaveta A. (University of Tartu, Tartu, Estonia). E-mail: volna@sibmail.com

Krasavin Dmitriy S. (Tomsk State University, Tomsk, Russian Federation). E-mail: iosfixed@gmail.com

Letner Oksana N. (Candidate of Physics and Mathematics, Tomsk State University, Tomsk, Russian Federation). E-mail: oksana.letner@gmail.com

Contribution of the authors: the authors contributed equally to this article. The authors declare no conflicts of interests.

Статья поступила в редакцию 30.12.2021; принята к публикации 22.03.2022

The article was submitted 30.12.2021; accepted for publication 22.03.2022