A Deep Learning Approach to Anomaly Detection in the Gaia Space Mission Data

Chapter · May 2019

DOI: 10.1007/978-3-030-20518-8_33

Marco Roberti
Università degli Studi di Torino
8 PUBLICATIONS **37** CITATIONS
SEE PROFILE

Mario Gai
National Institute of Astrophysics
217 PUBLICATIONS **10,793** CITATIONS
SEE PROFILE

A Deep Learning Approach to Anomaly Detection in the Gaia Space Mission Data

Alessandro Druetto[1], Marco Roberti[1], Rossella Cancelliere[1(✉)],
Davide Cavagnino[1], and Mario Gai[2]

[1] Computer Science Department, University of Turin,
Via Pessinetto 12, 10149 Torino, Italy
rossella.cancelliere@unito.it

[2] National Institute for Astrophysics, Astrophysical Observatory of Turin,
V. Osservatorio 20, 10025 Pino Torinese, Italy

Abstract. The data reduction system of the Gaia space mission generates a large amount of intermediate data and plots for diagnostics, beyond practical possibility of full human evaluation. We investigate the feasibility of adoption of deep learning tools for automatic detection of data anomalies, focusing on convolutional neural networks and comparing with a multilayer perceptron. The results evidence very good accuracy (∼99.7%) in the classification of the selected anomalies.

Keywords: Deep learning · Astronomical data · Diagnostics · Big data

1 Introduction

In recent years supervised learning's popularity increased dramatically, focusing on deep learning algorithms that are able to exploit large datasets [18]: in particular, deep learning has shown outstanding performance in spatio-temporal sequences processing (such as text or speech) and image processing. This led to two different – but equally successful – deep architectures: recurrent neural networks (RNNs, [29]) for tasks characterized by the presence of time sequences, and convolutional neural networks (CNNs) for imaging.

CNNs were firstly introduced in 1989 to recognize handwritten ZIP codes [23] and then used over the next ten years even if the relatively small datasets at the time were not suitable to proper training of CNNs with a huge number of parameters.

Only the advent of modern, much larger datasets makes the training of very deep CNNs effective: the breakthrough came in 2012, when Krizhevsky et al. [22] achieved the highest classification accuracy in the ILSVRC 2012 competition, using a CNN trained on the images of ImageNet dataset.

This radical shift would not have been possible without two factors that have been a real booster for its success, albeit not directly related to the field: on the

I. Rojas et al. (Eds.): IWANN 2019, LNCS 11507, pp. 390–401, 2019.
https://doi.org/10.1007/978-3-030-20518-8_33

one hand, the already mentioned impressive growth of data availability, mostly due to the development and the widespread diffusion of the internet technology; on the other hand, a computing capability hardly conceivable before, that takes advantage of recent years' parallelisation trend thanks to GPU technologies. These ideas and practices are now recognised as the foundation for modern CNNs. Since then, CNNs have been successfully applied, inter alia, in the recognition and classification of various items, such as hand-written digits (as in the MNIST [24] dataset), traffic signs [30], and more recently the 1000-category ImageNet dataset [22]; other important applications include object detection, image segmentation and motion detection.

CNNs have also recently found increasing usage in astrophysical applications, for better exploitation and inter-calibration of the large datasets produced from modern sky surveys' information. Some relevant examples include the development of CNNs for:

- derivation of fundamental stellar parameters (i.e. effective temperature, surface gravity and metallicity) [21];
- studies of galaxy morphology [32];
- high-resolution spectroscopic analysis using APO Galactic Evolution Experiment data [25];
- determination of positions and sizes of craters from Lunar digital elevation maps [31].

Also, in [34] ExoGAN (Exoplanet Generative Adversarial Network) is presented, a new deep-learning algorithm able to recognize molecular features, atmospheric trace-gas abundances, and planetary parameters using unsupervised learning.

In this paper we investigate the use of CNNs in the framework of the Gaia mission [28] of the European Space Agency (ESA), which will provide an all-sky catalogue of position, proper motion and parallax of about 1.7 billion objects among Milky Way stars and bright galaxies.

Our research concerns initial exploration of deep learning tools for data mining on the huge set of as yet unexploited Gaia plots, with the goal of improving on the identification of transients and peculiar operating conditions.

The current preliminary study is focused on two specific areas: (i) identification of runaway conditions on the Gaia plots (with parameters drifting beyond appropriate limiting values), and (ii) identification of one or more missing data in the plots.

In Sect. 2 we recall the main features of the Gaia mission and of the data used in this work; Sect. 3 includes a description of the deep architecture we use and of the experimental framework; we also analyse and evaluate the achieved results. Finally, in Sect. 4 we draw our conclusions, also outlining options for future work.

2 An Overview of Gaia

The Gaia mission will observe every object in the sky brighter than its limiting magnitude $V = 20\ mag$, with unprecedented astrometric accuracy [16,26] and astrophysical potential [12,14,17]. The current version of the output catalogue, based on the first half of the mission, is the Data Release 2 [15] (described also in https://cosmos.esa.int/web/gaia/dr2), available e.g. through the user interface https://gea.esac.esa.int/archive/, and it is already widely used by the astronomical community. The catalogue is materialised in the astrometric parameters (position, proper motion and parallax) of the sample of observed objects in DR2, mainly stars in our Galaxy. Gaia was launched on 2013, Dec. 19 from the French Guyana space center, and it reached its operating site (the L2 Lagrange point) about two weeks later. The five year mission lifetime has recently been extended by one plus one years.

The Gaia concept [28] relies on simultaneous observation of two fields of view, by two nominally equal telescopes, separated by a large basic angle ($\sim$106°), repeatedly covering the full sky by the combination of orbital revolution, satellite spin and precession. The Gaia focal plane has 7×9 large format astrometric Charge Coupled Device (CCD) sensors, complemented by specialised CCDs for detection of object as they enter the field, and sensors for photometric and spectroscopic measurements, for a grand total of about 1 Gpixel.

The data reduction scheme [26] derives, by self-consistency of the set of measurements, the kynematic information on celestial objects throughout the mission lifetime, factoring out the instrument parameters and their evolution by calibration.

The Gaia data reduction is managed by the Data Processing and Analysis Consortium (DPAC), including more than 450 European scientists. The DPAC is composed of Coordination Units (CUs), each in charge of specific parts of the overall reduction chain. Initial processing (Initial Data Treatment, IDT) performs preliminary estimate of several parameters (star position and magnitude, and initial attitude), which are then fed to different computing chains, all accessing the main database which includes raw and intermediate data, as well as the final results. Processing are split in many layers, operating on different data amounts and complexity, from daily operations (which must keep up with the continuous data inflow), up to the six-month cycle related to full sphere solution. Reduction software updates, required to account for unforeseen aspects of operating conditions, and for a progressively improving understanding of the instrument response, are also synchronised to the six month cycle.

The CU3, in particular, takes care of the so-called Core Processing, operating on the unpacked, filtered and pre-processed data of a large subset of well-behaved stars, and reconstructing their astrometric parameters, which are progressively improved for each object as the measurements pile up, by means of increasingly accurate (and more computer intensive) algorithms provided by the DPAC scientists.

The iterative astrometric core solution provides the calibration data and attitude reconstruction needed for all the other treatments, in addition to the astrometric solution of several million primary sources and the overall reference system of coordinates.

The whole Gaia database, including raw, intermediate and final data for the nominal five year mission, is estimated to exceed 1 Petabyte.

The processing chain includes a number of diagnostics functions, implemented in each unit at different levels, which have been used to monitor the mission behaviour in normal periods, the progressive evolution of many instrument parameters, and the insurgence of critical conditions related either to excessive variation of the payload (e.g. optical transmission degradation by contamination), or to external disturbances (e.g. solar flares), requiring modification of on-board setup. Moderate instrument variations must be taken into account by the data reduction system, by appropriate update of the parameters used, or by further algorithm development.

Diagnostics [9,10] is based on a large number of intermediate quantities, whose monitoring is often automated, but for cases where human assessment is considered as potentially necessary the data reduction system includes automatic generation of several thousand plots on a daily timeframe, which are stored in the database as intermediate data in graphical file format.

During critical periods, many such plots are studied by the payload experts for better understanding of the disturbances in play, and to define corrective measures if needed. Over most of the Gaia lifetime, fortunately, good operating conditions have been experienced so that most of the plots were not further considered. However, it is becoming clear that the payload is in a state of continual evolution, at the targeted precision level, and that the final mission accuracy will benefit of further improvements in the data processing taking into account the instrument response at a more detailed level.

In this paper, we evaluate the feasibility of using CNNs for processing of the huge set of as yet unexploited Gaia plots, with the goal of improving on the identification of transients and peculiar operating conditions.

We focus on two specific areas: identification of **runaway conditions** on the Gaia plots (with parameters drifting beyond appropriate limiting values), and **missing data** in the plots (ignored in the normal processing if it has small duration).

2.1 Selected Input Data

The first feasibility tests have been performed on the family of plots showing the daily statistics of along scan photo-center estimate. Such plots are generated on every day of operation for each of the 7×9 CCDs in the astrometric focal plane, and for each telescope. Since the electro-optical instrument response changes over the field of view, they are similar but appreciably different at the level of precision of Gaia. The data of each 30 min observation segment provide an average value and an error bar, due not only to photon statistics fluctuations, but also to "cosmic scatter", i.e. different characteristics of the many thousand detected celestial sources. Each plot includes therefore 48 points with errors. The abscissa is the mission running time, in satellite revolutions, whereas the vertical axis is in micro-meters with respect to the center of the readout window of each object. One detector pixel is 10 μm; a slip by more than one pixel in either direction of the average photo-center requires re-adjustment of the on-board parameters used for computation of the placement of the read-out windows.

To issue an alert and trigger the corrective action it is therefore necessary to implement an automatic detection of such conditions, so far managed mostly by human supervision. Similar checks, with different thresholds, can be applied to other relevant quantities summarising the quality of the measured data, e.g. the statistics of image moments like root mean square width, skewness and kurtosis.

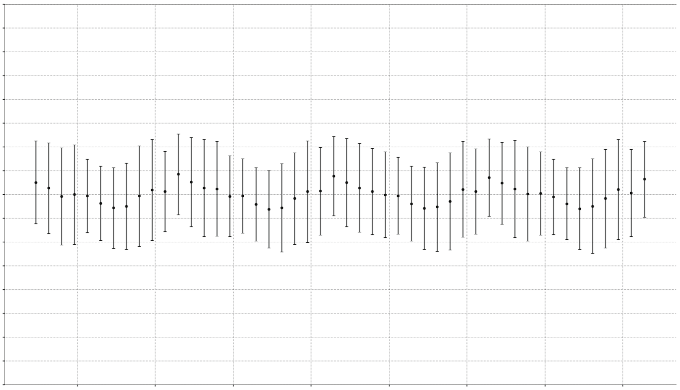
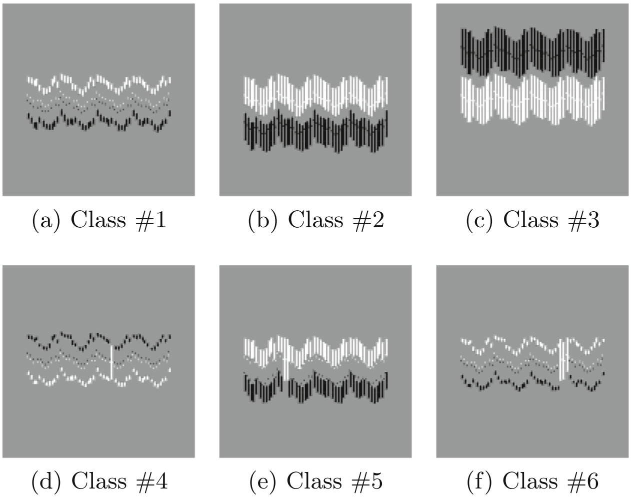


Fig. 1. A sample Gaia plot.

The original plots, generated for human evaluation, have format 1500 × 927 pixels, therefore imposing a large computational load due to their size. However, human readability requires a large amount of white space between points, which are placed in fixed positions (every half hour, over one day). Besides, the vertical labels are the same within each family of plots, and the horizontal labels (corresponding to the running time, a known quantity) are different but irrelevant to the test goals. Therefore, we decided to alleviate the data size, and correspondingly the computational load, by implementing an automated procedure which "squeezes" the initial plots by cutting the image strips associated to the labels, and removing most of the white space between useful data points. An example is shown in Fig. 1. We applied the same pre-processing also to a large number of plots expressly generated for simulating additional anomalous data; this is necessary for proper supervised CNN training, because "unfortunately" Gaia works in good operating conditions for most of the time.

Furthermore, in order to ease the detection of runaway conditions and missing data, we generated a new set of **difference plots** by subtracting from each of them the reference zero-offset case. This operation also removes the grid and axes of the original plots. The resulting images have much smaller format (128 × 128), but retain the initial information, providing a compression which is not strictly



(a) Class #1
(b) Class #2
(c) Class #3
(d) Class #4
(e) Class #5
(f) Class #6

Fig. 2. Examples for the six classes of input images.

required by the deep learning tools used for subsequent computation, but follows good general practices of economy.

These final images, shown in Fig. 2, are the inputs to the CNN models used for our diagnostic task.

3 Experimentation and Results

In this section we present in detail the neural architectures used during the experimentation and the obtained results.

3.1 Our Proposed Model

Our principal aim is to classify GAIA images with respect to the presence or absence of certain kind of anomalies. In the task of supervised classification, there exist a lot of successful approaches: two widespread examples are support vector machines and random forests.

A support vector machine [13], a binary classifier in its standard formulation, builds a special kind of separation rule, called a linear classifier, with theoretical guarantees of good predictive performance. Statistical learning theory [33] gives theoretical basis for this family of methods. To work even with non-linear data, the so-called kernel trick can be used to construct special kinds of non-linear rules. Also, many approaches exist to build a non-binary classifier system from a set of binary classifiers (one-vs-all, one-vs-one, error correcting output codes (ECOC) [11], Directed Acyclic Graph (DAG) [27]). In all of these approaches we combine prediction results from multiple previously trained binary classifiers.

A random forest [5] is a machine learning technique useful for prediction problems. The seminal algorithm, developed by Breiman [8], applies random feature selection [2,19] and bootstrap aggregation [7] to a set of classification trees to obtain a better prediction. It is known that decision trees are not the most efficient classification method, as they are highly unstable and very likely to overfit training data. However, random forest mitigates individual trees overfitting [3,4,6] by a voting system over a set of separately trained decision trees.

Taking into account these approaches, we decided to experiment CNNs over GAIA dataset, verifying also the applicability of random forests and support vector machines.

In Fig. 2 we can see examples of all six classes. Class #1 (Fig. 2a) contains images not evidencing anomalies. One typical anomaly in the plot is the **runaway condition**: classes #2 (Fig. 2b) and #3 (Fig. 2c) represent respectively the cases of downward and upward shift of the data points. The other anomaly we want to investigate corresponds to one or more consecutive **missing data** points, resulting in adjacent vertical white lines: classes #4 (Fig. 2d), #5 (Fig. 2e) and #6 (Fig. 2f) identify one, two or three consecutive missing data points respectively. Wider gaps, corresponding to more relevant on-board failures, are detected by other subsystems triggering suitable corrective actions.

The difference plots are composed by vertical strokes in greyscale; those are comparable to handwritten strokes produced when someone draws or writes something. This fact gives us a hint to analyse our data with an architecture similar to the one used as state-of-art for the MNIST dataset of handwritten digits, i.e. the CNN with 2 convolutional blocks (*double-convolution CNN*) described in the following.

Recalling briefly how a CNN is usually built, its principal constituents are a set of convolutional blocks, followed by a set of fully connected layers. A convolutional block is composed by some layers of convolutional filters, followed by an activation function and a pooling.

A convolution acts as a filter that multiplies each pixel in the $N \times N$ subregion for the corresponding value, summing up all values to get a single scalar.

The previously computed weighted sum is then fed to an activation function, usually the Rectified Linear Unit, or ReLU.

After application of both the filter and the activation, a pooling is performed: every square block of size $M \times M$ (typically $M < N$) is represented by its maximum or average value.

The set of fully connected layers behaves as a classical multilayer perceptron; all neurons of the last convolutional layer are connected to one (or more) layers of hidden neurons, and from here to the output layer.

Our proposed double-convolution CNN is shown in Fig. 3.

In the perspective of possible reduction of the computational effort in operation, we also explored a *single-convolution CNN* containing only one convolutional block.

We decided to compare our CNNs also to a simpler network structure, in particular our third model is a *multilayer perceptron* (MLP).

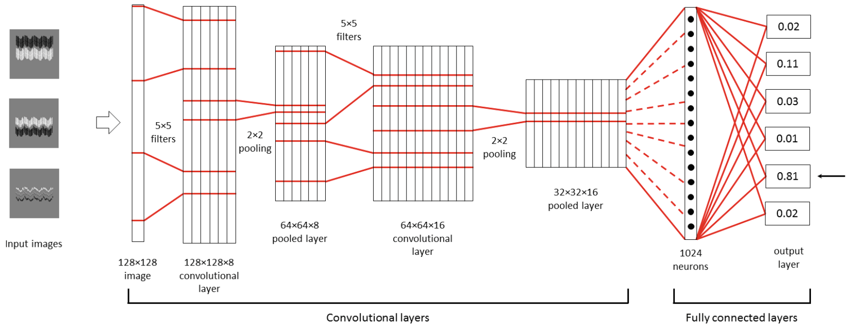


Fig. 3. CNN structure.

For analogy with the previously built CNN, the MLP structure is extracted directly from such network. In particular, we used the final fully connected part of the CNNs. All models share the choice of the ReLU activation function.

For the entire validation and testing processes, we generated a total of 9000 images: 6000 for the training set, 1500 for the validation set and 1500 for the test set. All such images are equally spread among the 6 different classes, thus providing a balanced distribution.

Since the possible classes are 6, the typical representation of labels for each image is a one-hot mono-dimensional array. This array can represent, in fact, a probability distribution and works well with the cross-entropy loss function. We minimize such loss using the ADAM optimizer [20].

The previously introduced validation set guided the tuning of hyperparameters in our models. In particular, we grid-searched for best values over the tunable parameters: convolutional filters size; number of convolutional filters; max-pooling filter size; dropout rate; batch size; and, both in the MLP and in the fully connected layer on the CNNs, number of hidden neurons.

Table 1. Final hyperparameter choice.

Hyperparameter	CNN model	
	Single-convolution	Double-convolution
1st conv. filters size	5×5	5×5
2nd conv. filters size	-	5×5
1st conv. number of filters	16	8
2nd conv. number of filters	-	16
Max-pooling filter size	2×2	2×2
Dropout rate	0.5	0.5
Batch size	50	50
Number of hidden neurons	1024	1024

The best CNN structures appear to be the double-convolution one with 8 and 16 filters respectively in the two convolutional blocks, and the single-convolution one with 16 filters. In both cases, 5 × 5 filters obtain the best results.

Table 1 shows the optimal hyperparameter values.

The better-performing MLP network is the one with 200 hidden neurons, with acceptably good results between 100 and 400.

All tests are performed with 10 different random weights initializations in order to yield different starting points to the training process, hence providing statistics (i.e. average values and standard deviations) to our tests. The stop criterion used during training is the loss stabilization.

3.2 Experimental Results

All the experiments are run using a software framework based on TensorFlow [1] and written in `Python 3.6`.

The three models, selected during validation procedure, are tested on a GeForce GTX 1070 with 1920 CUDA cores.

Table 2. Test results: statistics over 10 different random weights initializations.

	Accuracy		Time (s)	
	Avg	Std	Avg	Std
Single-convolution CNN	0.9967	0.0045	158.64	0.12
Double-convolution CNN	0.9965	0.0013	127.56	0.07
MLP	0.9655	0.0110	18.54	0.05

The results achieved on the test set are summarised in Table 2. We remark that both CNNs achieve almost the same accuracy, over 99%; however, the double-convolution CNN seems more stable, since the standard deviation is smaller. MLP reaches an accuracy ∼3% lower than CNNs: this might suggest that a non-convolutional network is less able to evince structure and discrepancies between classes than a convolutional one. Besides, MLP standard deviation is more than two times higher than CNN's worst case. We also remark that its training time is smaller, as expected.

Notwithstanding the excellent performance achieved by the CNN approach, for the sake of completeness we also explored the aforementioned methods, i.e. random forest and support vector machine. The former reached a test accuracy of 0.9387 with 4096 trees; the latter, instead, provided a test accuracy of 0.5093 with RBF kernel and $C = 1$. We suppose that the poor performance of support vector machine is due to the extremely high dimensionality of input data, since each image pixel is an independent dimension.

In order to assess the statistical significance of our results, we decided to perform a Student's t-test. It provides evidence, at a significance level of 99.9%, that the MLP performance is worse than either CNN architectures. Similarly, the performance of single-convolution CNN and double-convolution CNN is equal, at the same significance level.

An interesting fact is that the CNN training time is smaller in the double-convolution case. This behavior is due to the presence, in this case, of a second pooling layer further reducing the input size of the fully connected layer.

4 Conclusions

We deal with the issue of detection and classification of anomalous data on automatically generated images from the intermediate processing in the data reduction system of the Gaia space mission.

We investigate the application of convolutional neural networks, evidencing very good classification accuracy and quite acceptable training time.

Single- and double-convolution CNNs have comparable performance, with better stability and shorter training time in the latter case. However, the former, lighter architecture would result in faster runtime on operation. MLP still provides good classification performance, but significantly lower than either CNN (in spite of faster training and running time). Random forests and support vector machines achieve, respectively, acceptable and poor results.

The results are promising with respect to possible adoption of CNNs and deep learning tools in the Gaia data reduction system. Further investigation may be devoted to increasing the range of target anomalies, thus refining the diagnostic class definition.

Acknowledgements. We acknowledge the contribution of sample plots and discussion on the requirements from D. Busonero and E. Licata (INAF-OATo). The activity has been partially funded by the Italian Space Agency (ASI) under contracts Gaia Mission, The Italian Participation to DPAC, 2014-025-R.1.2015 and 2018-24-HH.0.

References

1. Abadi, M., et al.: TensorFlow: a system for large-scale machine learning. In: Keeton, K., Roscoe, T. (eds.) 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016, Savannah, GA, USA, 2–4 November 2016, pp. 265–283. USENIX Association (2016)
2. Amit, Y., Geman, D.: Shape quantization and recognition with randomized trees. Neural Comput. **9**(7), 1545–1588 (1997)
3. Banfield, R.E., Hall, L.O., Bowyer, K.W., Bhadoria, D., Kegelmeyer, W.P., Eschrich, S.: A comparison of ensemble creation techniques. In: Roli, F., Kittler, J., Windeatt, T. (eds.) MCS 2004. LNCS, vol. 3077, pp. 223–232. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-25966-4_22
4. Bernard, S., Heutte, L., Adam, S.: On the selection of decision trees in Random Forests. In: 2009 International Joint Conference on Neural Networks, pp. 302–307 (2009)
5. Bharathidason, S.: Improving classification accuracy based on random forest model with uncorrelated high performing trees (2014)

6. Boinee, P., Angelis, R.D., Foresti, G.L.: Ensembling classifiers – an application to image data classification from cherenkov telescope experiment (2005)
7. Breiman, L.: Heuristics of instability and stabilization in model selection. Ann. Stat. **24**(6), 2350–2383 (1996)
8. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
9. Busonero, D., Lattanzi, M., Gai, M., Licata, E., Messineo, R.: Running AIM: initial data treatment and μ-arcsec level calibration procedures for Gaia within the astrometric verification unit. In: Modeling, Systems Engineering, and Project Management for Astronomy VI, p. 91500K (2014)
10. Busonero, D., Licata, E., Gai, M.: Astrometric instrument model software tool for Gaia real-time instrument health monitoring and diagnostic. Revista Mexicana de Astronomía y Astrofísica **45**, 39–42 (2014)
11. Dietterich, T.G., Bakiri, G.: Solving multiclass learning problems via error- correcting output codes. CoRR cs.AI/9501101 (1995)
12. Evans, D., et al.: Gaia data release 2-photometric content and validation. Astron. Astrophys. **616**, A4 (2018)
13. Fradkin, D., Muchnik, I.: Support Vector Machines for Classification. DIMACS Series in Discrete Mathematics and Theoretical Computer Science (2006)
14. Gaia Collaboration, Babusiaux, C., et al.: Gaia data release 2. Observational Hertzsprung-Russell diagrams. Astron. Astrophys. **616**, A10 (2018)
15. Gaia Collaboration, Brown, A.G.A., et al.: Gaia data release 2. Summary of the contents and survey properties. Astron. Astrophys. **616**, A1 (2018)
16. Gaia Collaboration, Mignard, F., et al.: Gaia data release 2. The celestial reference frame (Gaia-CRF2). Astron. Astrophys. **616**, A14 (2018)
17. Gaia Collaboration, Spoto, F., et al.: Gaia data release 2. Observations of solar system objects. Astron. Astrophys. **616**, A13 (2018)
18. Goodfellow, I.J., Bengio, Y., Courville, A.C.: Deep Learning. MIT Press, Cambridge (2016)
19. Ho, T.K.: The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. **20**(8), 832–844 (1998)
20. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. CoRR abs/1412.6980 (2014)
21. Kou, R., Petit, P., Paletou, F., Kulenthirarajah, L., Glorian, J.-M.: Deep learning determination of stellar atmospheric fundamental parameters. In: Proceedings of the Annual meeting of the French Society of Astronomy and Astrophysics, SF2A-2018, pp. 167–169 (2018)
22. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, NIPS 2012, Lake Tahoe, Nevada, pp. 1097–1105. Curran Associates Inc. (2012)
23. LeCun, Y., et al.: Backpropagation applied to handwritten zip code recognition. Neural Comput. **1**(4), 541–551 (1989)
24. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
25. Leung, H.W., Bovy, J.: Deep learning of multi-element abundances from high resolution spectroscopic data. Mon. Not. R. Astron. Soc. **483**, 3255–3277 (2019)
26. Lindegren, L., et al.: Gaia data release 2. The astrometric solution. Astron. Astrophys. **616**, A2 (2018)
27. Platt, J.C., Cristianini, N., Shawe-Taylor, J.: Large margin DAGs for multiclass classification (2000)

28. Prusti, T., et al.: The Gaia mission. Astron. Astrophys. **595**, A1 (2016)
29. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986)
30. Sermanet, P., LeCun, Y.: Traffic sign recognition with multi-scale Convolutional Networks. In: The 2011 International Joint Conference on Neural Networks, pp. 2809–2813 (2011)
31. Silburt, A., et al.: Lunar crater identification via deep learning. Icarus **317**, 27–38 (2019)
32. Tuccillo, D., Huertas-Company, M., Decencière, E., Velasco-Forero, S., Domínguez Sánchez, H., Dimauro, P.: Deep learning for galaxy surface brightness profile fitting. Mon. Not. R. Astron. Soc. **475**, 894–909 (2018)
33. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, Heidelberg (1995). https://doi.org/10.1007/978-1-4757-2440-0
34. Zingales, T., Waldmann, I.P.: ExoGAN: retrieving exoplanetary atmospheres using deep convolutional generative adversarial networks. Astron. J. **156**, 268 (2018)