Neural networks approaches to star cluster studies:

Artificial neural networks, convolutional neural networks and mixed data models

Duarte Gerardo Branco Nº54502
João Lucas Figueiredo Ferraz Nº49420

Advanced machine learning, May 2023

Faculdade de Ciências da Universidade de Lisboa

Introduction

In this project we used supervised deep learning to train models on the task of associating isochrone parameters to star clusters. Isochrones consist of curves representing a population of stars with the same age but different masses. Fitting isochrones manually is a time consuming process and model dependent since isochrones come from simulated models. Therefore, we aim in our project to automate the process first by attempting to replicade an already done attempt [1] using an artificial neural network (ANN) then attempting to use a convolutional neural network (CNN) and a mixed data model (MDM) to improve upon the original model. For this, we trained models using star cluster information available in [2] and by using the photometric information available for the stars in each cluster, the cluster's age, distance and reddening, the models became able to determine cluster characteristics.

Data

The dataset used to train the machines in this paper is composed of two tables: a table with cluster data and a table with star data for the stars that compose each cluster. The tables were taken from the same catalogue used in [1] (catalogue available on [2]).

From the catalogue the target labels are the cluster's age in log scale (AgeNN), extinction in the V band (AVNN) and distance in kilo parsec (calculated using the $10^{\frac{DMNN+5}{5}}/1000$).

The dataset contains 1867 usable clusters and much like in [1] we augmented the dataset in a similar manner to double the amount of clusters and evaluate the effect of this augmented set when used for training. The data was augmented the same way as chapter 3.2 of [1] with the exception that the upper limit for the simulated distance modulus was 10 and the upper limit for the extinction in visual was 3. One of the main data inputs was a Hertzsprung-Russell (HR) diagram using Gmag as the magnitude (y) value and BP-RP as the colour (x) value.

The biggest issue when creating these models stems from the fact that three quantities are being regressed, and as such if one of them is closer to its intended target than the others the model will seem to be performing better than it actually is. Therefore the label outputs were all scaled using the MinMaxScaler from sklearn.

Artificial neural network

Implementation

First, we attempted to replicate the model in [1]. The architecture of the model described in the article is shown in figure 6, it has 410 inputs that are the results of applying a 2D histogram to the HR diagram figures and normalizing the count of points that land in each bin. In essence this pixelizes the image so it can be inputted into the ANN. Then there are three more inputs which are: the slope in the relation between

colour and magnitude for the stars whose distance-corrected magnitude is brighter than 4; the mean colour of stars whose distance-corrected magnitude is between 4 and 5; the median parallax. These three inputs work as guesses for the three outputs which are the cluster's age, extinction in the V band and its distance.

Not much information was given about the inputs in the article but we generated the plots as close as possible by limiting every image within 0 to 4 in the BP-RP (x axis) and 10 to 18 for the Gmag (y axis). Then divided the image into 410 sections each having a bin width of 0.1 in the x axis and a bin height of 0.8 in the y axis (figure 1).

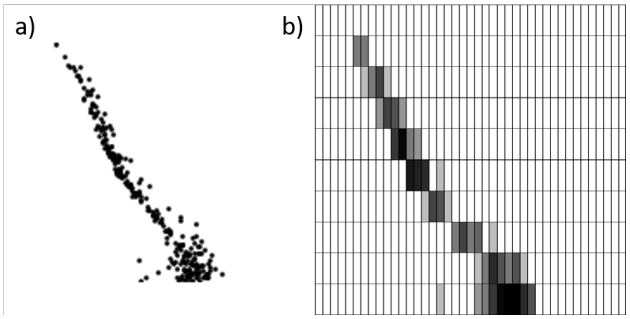

Figure 1: *a) image, b) image after being pixelated into 410 bins and the couints in each bin normalized. The cluster show is Alessi 5.*

Results

The ANN was then trained on the original 1867 clusters and the augmented data set of 3734 clusters. For both cases it was trained with: shuffling, 200 epochs, batch size equal to 299 for non augmented and 400 for augmented, mean squared error loss and adam optimizer.

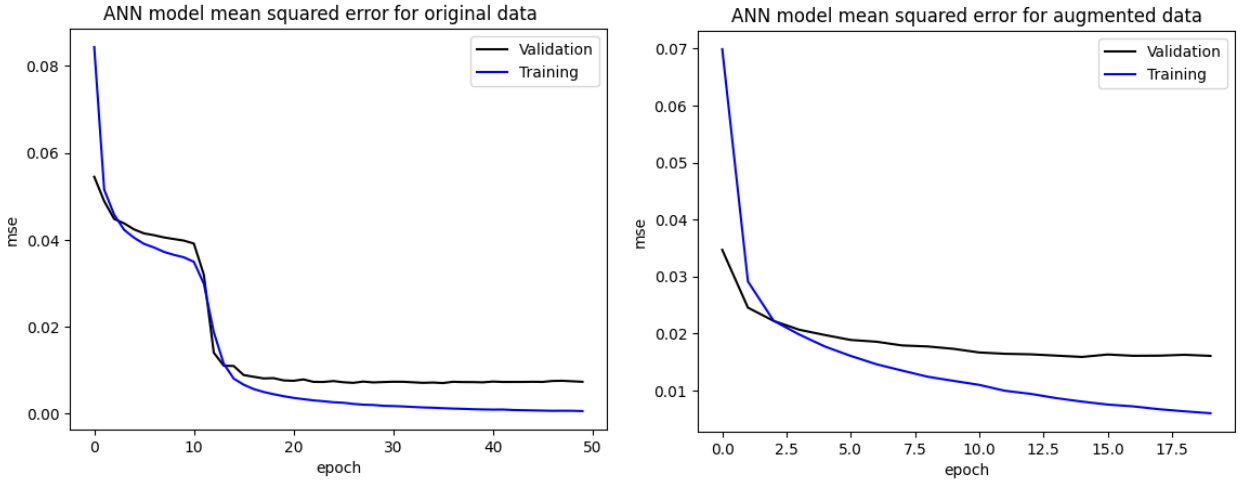

Figure 2: *Mean squared error in function of epoch for the ANN both on the original 1867 dataset and the augmented one.*

Observing figure 2 it is clear that when the value of the mean square error (MSE) for the validation data stagnates and deviates from that of the training that the network begins to become overfitted. From now on in this paper we will only show the best versions of the models. When comparing the augmented version with the original version it is clear that overfitting beings much sooner in the augmented data set, this is because the augmented dataset was trained with a higher batchsize, however, the best performing model was the one trained on the original dataset rather than the augmented one since the augmented model had MSE=0.334 and an R^2 score of 0.487 while the original set had MSE=0.176 and an R^2 score of 0.747

(check table 1 for comparisons between models). This indicates that the increase in batchsize to speed up training was possibly a wrong call or there was some characteristic, perhaps related to the histograms, that decreased performance after it was augmented.

Convolutional neural network

Implementation

Since in the paper the main input is a pixelated version of the hr diagram we attempted to use instead a convolutional neural network to test its effectiveness over the "pixelated" approach that loses a lot of detail. All images fed into the CNN were generated using matplotlib pyplot savefig and every figure was 4 by 4 inches with 180 by 180 pixels.

The architecture for our convolutional neural networks is shown in figure 5.

Results

The CNN was then trained in the same way as the ANN. First a 150 epoch fit revealed when any overfitting began to happen, then the best performing model was selected (figure 3).

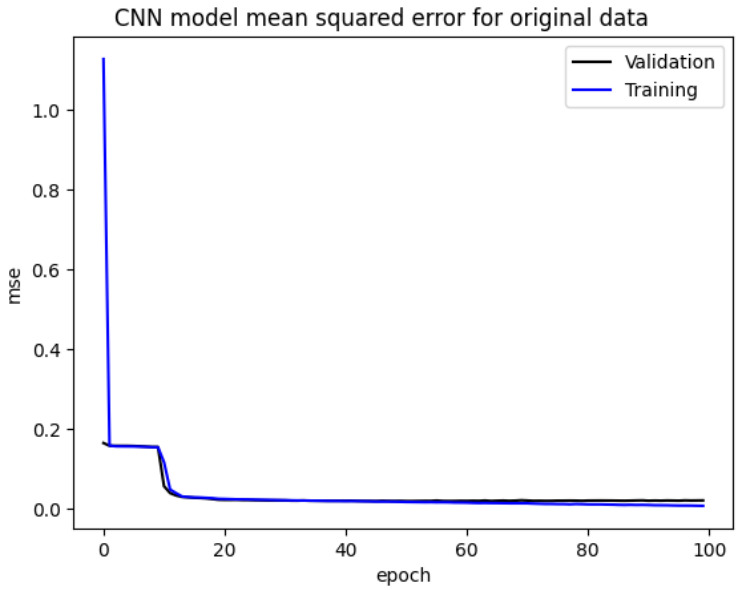

Figure 3: *Mean squared error in function of epoch for the CNN for the original 1867 dataset.*

Interestingly the CNN when compared to the ANN performed worse having an MSE=0.840 and an R^2 score of 0.242 (check table 1 for comparisons between models). We believe this was caused because the CNN has no guess input like the ANN and therefore requires a higher variety of examples to outperform the ANN. However when the model was trained with the augmented dataset it performed quite worse, this was most likely because we kept the augmentation parameters relatively stable and the variations do not cause many large changes in the images and unlike the ANN when an image starts to move outside the borders and get cut, there are no guess parameters to help the CNN still predict a correct result. As such creating a more varied augmented dataset by using larger possible parameter changes and by also creating copies of the clusters with less stars, thus leading to images with less points should help the CNN to achieve better results.

Mixed data model

Implementation

When first observing the architecture of the model from the article [1] it becomes apparent that a better way to implement the graphical analysis (the histogram input) while also not wasting any other data obtained by the survey (the guesses for the output) a mixed data model can be used, where a CNN analyses the HR diagram and an ANN analyses the three guesses. This way there is no need for loss of detail by pixelating the images in order to make it possible to do everything with a single ANN.

The architecture of our mixed data model was a combination of two branches, an ANN branch (figure 7) and a CNN branch (figure 8), both of these branches then combine and the structure then follows the same as the hidden layers of the ANN from the article. The full structure is shown in figure 9.

Results

The mixed data model was then compiled and trained in the same way as the previous models on the original 1867 dataset. Figure 4 shows the results of training. Even though there appears to be overfitting above 50 epochs the 100 epoch model performed better scoring a mean squared error of 0.167 on the predictions for the validation set and an R^2 score of 0.761 (check table 1 for comparisons between models).

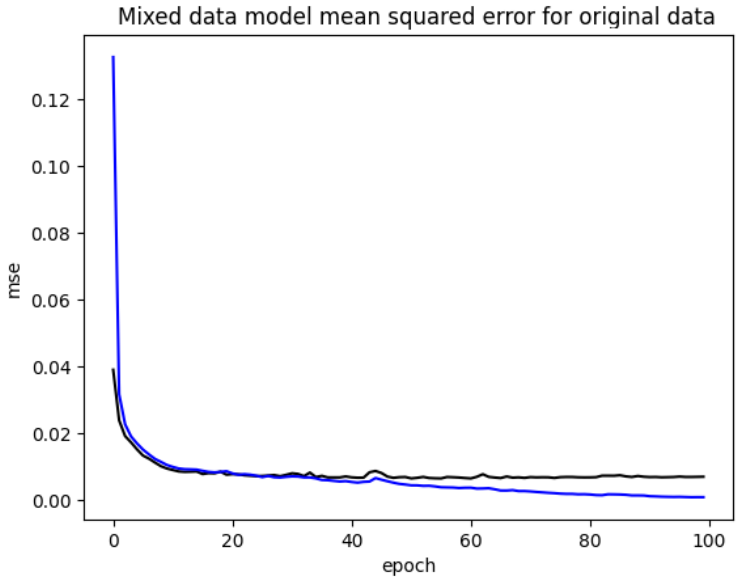

Figure 4: *Mean squared error in function of epoch for the MDM for the original 1867 dataset.*

When comparing this with the CNN the results are remarkably better. The model was also efficient enough in training not taking that long to train. It also outperformed the ANN ever so slightly. Like stated before the CNN performed poorly for a number of possible reasons and these same reasons also affect the MDM as such a larger dataset or better performed data augmentation should increase the performance of the MDM substantially.

This approach has a number of advantages over the model used in the article like its faster convergence time and the benefit of no complicated data pre-processing (no histogram required).

Table 1: *Table comparing the mean squared error and R^2 score of all three models tested.*

Model	Mean squared error	R^2 score
ANN	0.176	0.747
CNN	0.841	0.242
MDM	0.167	0.761

Conclusion

We attempted to replicate the model used in the article and were unsuccessful at obtaining similar results. However the results shown for the use of a single CNN and especially of the mixed data model make us confident that the model in the article could be improved if it was constructed as a mixed data model making use of a convolutional neural network to not only obtain more features from the HR diagram but to also enable the use for a larger scale for the diagram without the loss of information that comes from pixelating the image with a fixed bin size. We also conclude that the largest improvement that could be done to our models stems from the dataset or data augmentation and believe that with a larger dataset the results should be promising.

Model architectures

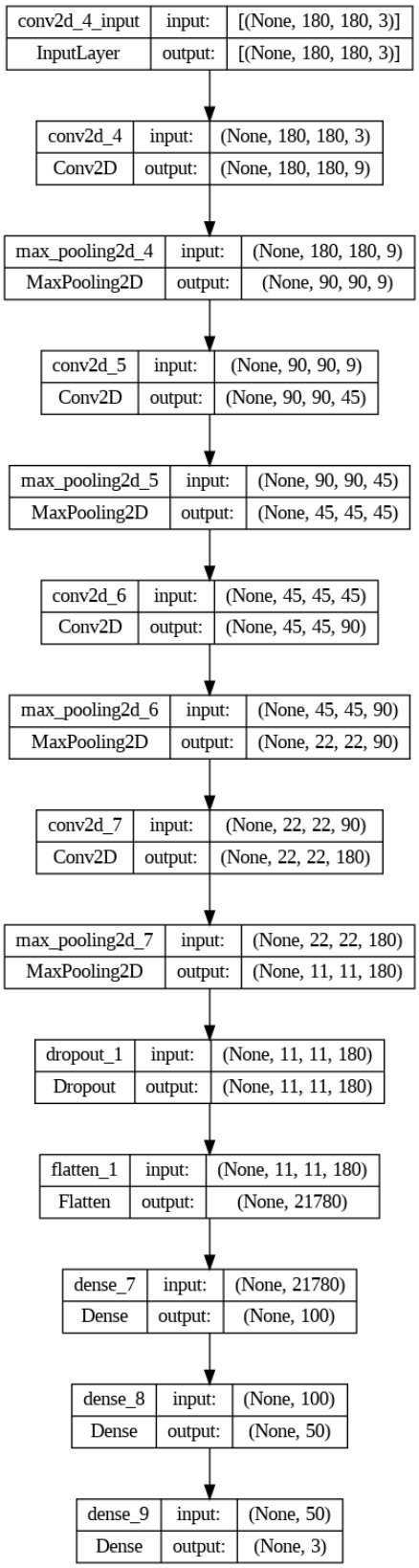

Figure 5: *Architecture of the CNN.*

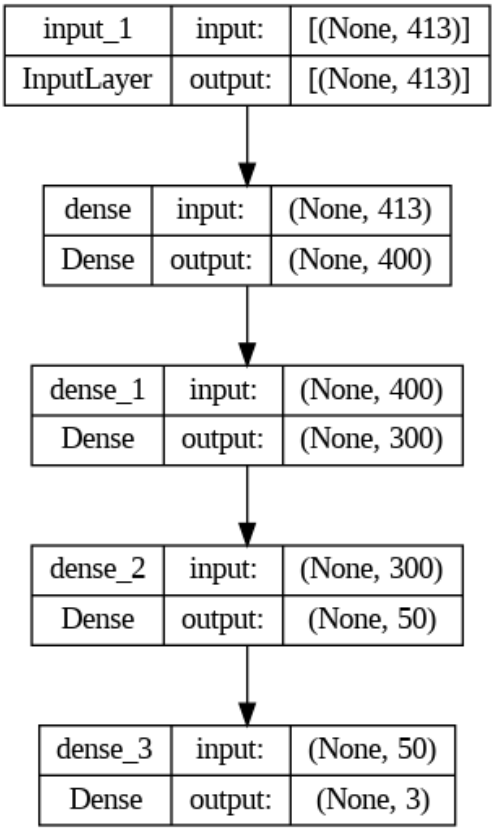

Figure 6: *Architecture of the ANN.*

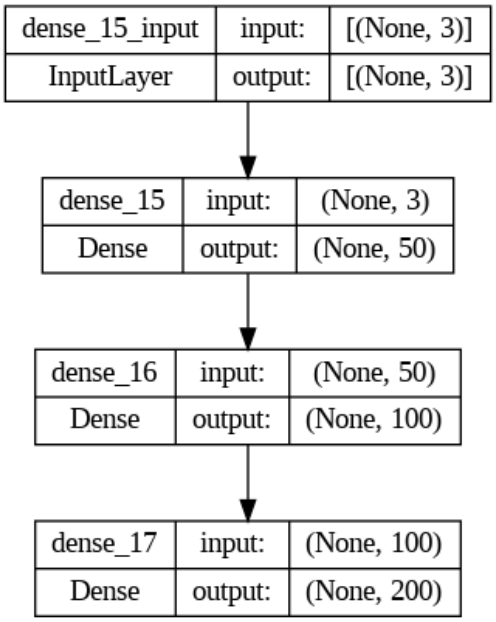

Figure 7: *Architecture of the ANN branch for the MDM.*

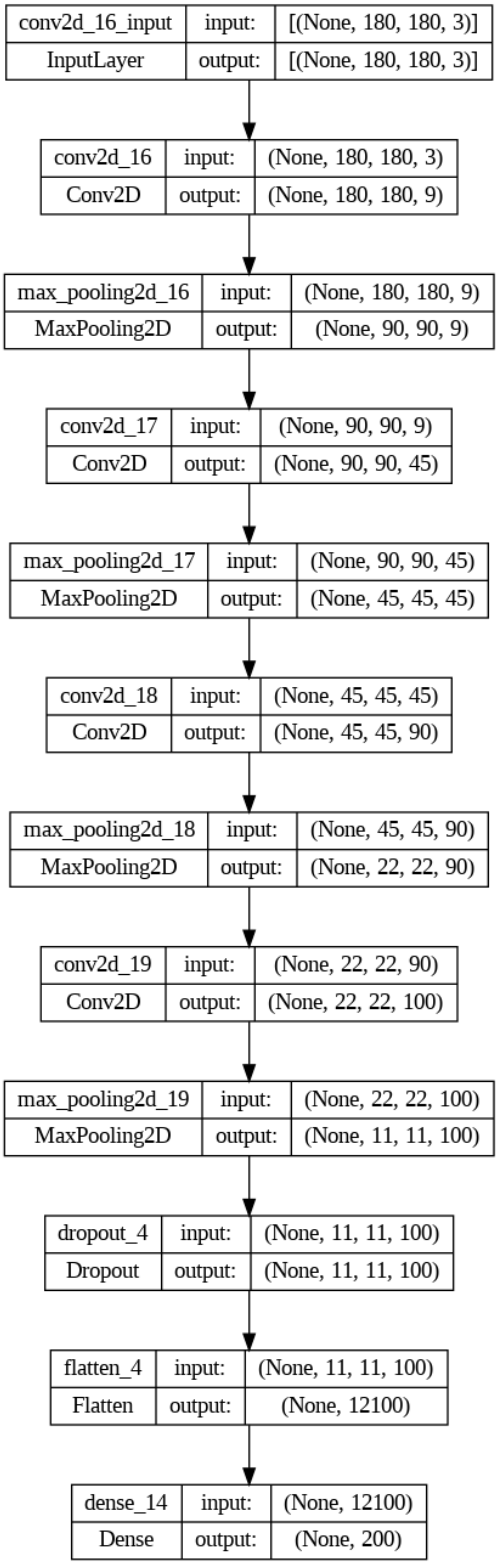

Figure 8: *Architecture of the CNN branch for the MDM.*

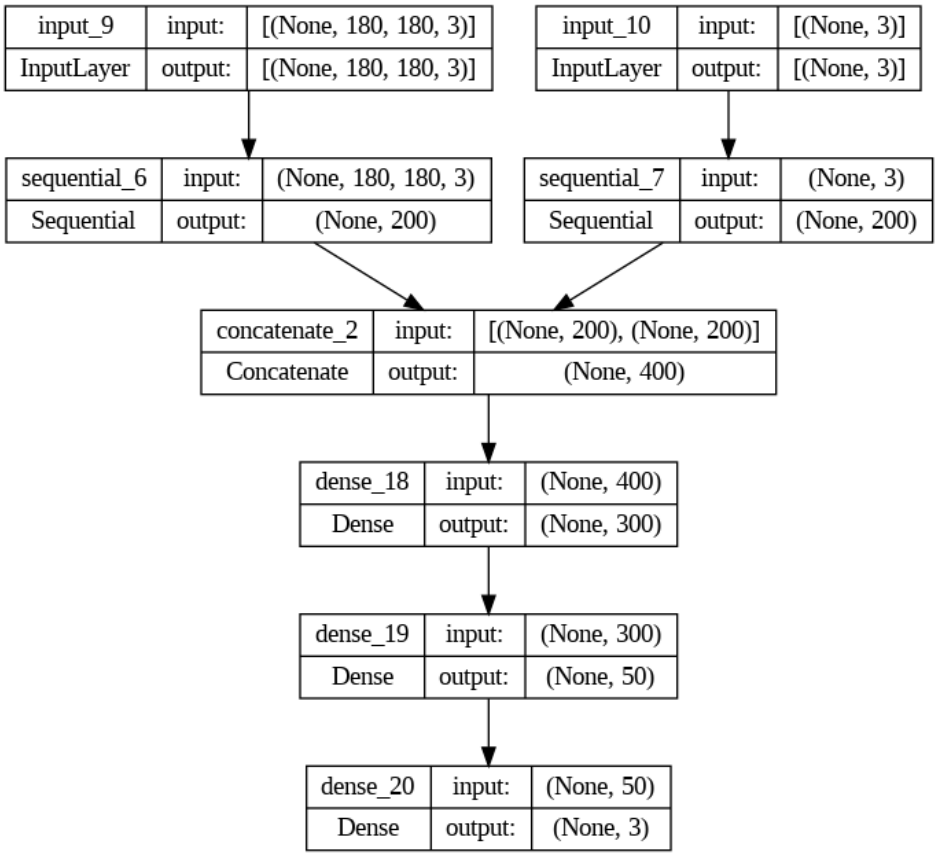

Figure 9: *Architecture of the MDM.*

References

[1] Cantat-Gaudin, T., Anders, F., Castro-Ginard, A., Jordi, C., Romero-Gómez, M., Soubiran, C., Casamiquela, L., Tarricq, Y., Moitinho, A., Vallenari, A., Bragaglia, A., Krone-Martins, A., and Kounkel, M. Painting a portrait of the galactic disc with its stellar clusters. *A&A*, 640:A1, 2020.

[2] Castro-Ginard A. Jordi C. Romero-Gomez M. Soubiran C. Casamiquela L. Tarricq Y. Moitinho A. Vallenari A. Bragaglia A. Krone-Martins A. Kounkel M. Cantat-Gaudin T., Anders F. Portrait galactic disc, 2020.