A&A 621, A13 (2019)
https://doi.org/10.1051/0004-6361/201732303

Astronomy & Astrophysics

Leaves on trees: identifying halo stars with extreme gradient boosted trees

Jovan Veljanoski, Amina Helmi, Maarten Breddels, and Lorenzo Posti

Kapteyn Astronomical Institute, University of Groningen, Landleven 12, 9747 AD Groningen, The Netherlands
e-mail: jovan@astro.rug.nl

Received 15 November 2017 / Accepted 18 April 2018

ABSTRACT

Context. Extended stellar haloes are a natural by-product of the hierarchical formation of massive galaxies like the Milky Way. If merging is a non-negligible factor in the growth of our Galaxy, evidence of such events should be encoded in its stellar halo. The reliable identification of genuine halo stars is a challenging task, however.
Aims. With the advent of the *Gaia* space telescope, we are ushered into a new era of Galactic astronomy. The first *Gaia* data release contains the positions, parallaxes, and proper motions for over two million stars, mostly in the solar neighbourhood. The second *Gaia* data release will enlarge this sample to over 1.5 billion stars, the brightest ∼5 million of which will have full phase-space information. Our aim for this paper is to develop a machine learning model for reliably identifying halo stars, even when their full phase-space information is not available.
Methods. We use the Gradient Boosted Trees algorithm to build a supervised halo star classifier. The classifier is trained on a sample of stars extracted from the *Gaia* Universe Model Snapshot, which is also convolved with the errors of the public TGAS data, which is a subset of *Gaia* DR1, as well as with the expected uncertainties for the upcoming *Gaia* DR2 catalogue. We also trained our classifier on a dataset resulting from the cross-match between the TGAS and RAVE catalogues, where the halo stars are labelled in an entirely model-independent way. We then use this model to identify halo stars in TGAS.
Results. When full phase-space information is available and for *Gaia* DR2-like uncertainties, our classifier is able to recover 90% of the halo stars with at most 30% distance errors, in a completely unseen test set and with negligible levels of contamination. When line-of-sight velocity is not available, we recover ∼60% of such halo stars, with less than 10% contamination. When applied to the TGAS catalogue, our classifier detects 337 high confidence red giant branch halo stars. At first glance this number may seem small, however, it is consistent with the expectation from the models, given the uncertainties in the data. The large parallax errors are in fact the biggest limitation in our ability to identify a large number of halo stars in all the cases studied.

Key words. Galaxy: halo – Galaxy: kinematics and dynamics

1. Introduction

Our current understanding of galaxy assembly is that it occurs in a hierarchical manner: smaller dark matter haloes merge to form larger, more massive objects. This process results in the formation of diffuse stellar haloes, which envelop massive galaxies like our Milky Way (e.g. Cooper et al. 2010; Helmi et al. 2011). Even though they comprise less than 1% of the total stellar mass of a galaxy, in principle, stellar haloes keep records of most past merger and accretion events that their host has experienced, encoded in the dynamics, chemistry, age, spatial structure, and star-formation history of their stellar populations. Thus, detailed studies of the stars comprising galactic haloes enable us to unravel the formation and evolution history of massive galaxies, and with that to test the ΛCDM paradigm.

Due to the close proximity, it is only natural that the most detailed studies of stellar haloes have been made in the Local Group. Deep, wide-field ground-based photometric surveys such as SDSS and PanSTARRS have been able to uncover a variety of stellar streams in the Milky Way (e.g. Newberg et al. 2002; Belokurov et al. 2006; Grillmair & Dionatos 2006; Grillmair 2006; Martin et al. 2014; Bernard et al. 2014, 2016; Balbinot et al. 2016). These are thought to be the remnants of tidally disrupted dwarf galaxies or star clusters, and thus can serve as markers of recent accretion events. Indeed, we are currently witnessing how the Sagittarius dwarf galaxy is being accreted onto the Milky Way (e.g. Ibata et al. 1994, 1995; Koposov et al. 2012; Slater et al. 2013), which serves as evidence that the Galactic halo is still actively evolving. The Milky Way is not unique in this respect. Stellar streams are also found in our closest massive neighbour, M31 (e.g. McConnachie et al. 2009; Ibata et al. 2014), as well as in several more distant galaxies (e.g. Martínez-Delgado et al. 2008, 2010; Crnojević et al. 2016), typically in their outer haloes.

In the Milky Way, the spatially coherent stellar streams are often used to constrain the enclosed mass and the shape of the underlying gravitational potential (e.g. Law et al. 2009; Law & Majewski 2010; Sanders & Binney 2013; Bovy et al. 2016). They do not paint the full picture, however. A significant fraction of the stars in the halo of our Galaxy also belong to the so-called smooth component. This component may be a remnant of the initial stages of the Milky Way's formation, and may contain information about the very first objects that built our galaxy. These primordial building blocks have likely sunk deeper into the Galactic potential well, and due to the shorter dynamical timescales are now likely to be fully phased-mixed and cannot be readily observed on the sky as coherent structures (Helmi & White 1999). In order to detect them, one needs to know accurately the three-dimensional (3D) positions and velocities of their constituent stars (e.g. Helmi & de Zeeuw 2000).

At least for now, the Milky Way is the only massive galaxy for which we can obtain sufficiently accurate astrophysical measurements of its stars, hoping to decipher its assembly history. With the advent of the *Gaia* satellite we enter a new era of Galactic astrophysics. The primary dataset of *Gaia* Data Release 1 (DR1) provides the positions, parallaxes, and mean proper motions for over two million stars in common with the *Tycho-2* and the HIPPARCOS catalogues, otherwise known as the *Tycho-Gaia* Astrometric Solution (TGAS). The secondary dataset of *Gaia* DR1 consists of on-sky positions for over one billion sources and their mean G-band magnitudes (Gaia Collaboration 2016). The biggest step forward will happen with the second data release (DR2), scheduled for the end of April 2018, which will contain 3D positions, proper motions, and optical photometry (G, $G_{\rm BP}$, $G_{\rm RP}$) for over one billion stars. For the brightest $\sim 3-5$ million of those, full phase-space information will be available. This will enable us to dig deeper into our Galaxy's history than ever before.

Before being able to explore the Milky Way's past, we need a method of identifying halo stars. In the pre-*Gaia* era, candidate halo stars were selected based on their proper motions and metallicities (e.g. Majewski 1992; Carney et al. 1996; Majewski et al. 1996; Smith et al. 2009). Such samples may be incomplete since they are biased against stars having small proper motions. More recently, as surveys became wider and deeper, studies have also focused on very distant stars in order to constrain the properties of the halo outside the solar neighbourhood. Commonly used halo tracers are RR Lyrae (e.g Watkins et al. 2009; Sesar et al. 2010; Drake et al. 2013a,b; Torrealba et al. 2015) , which can be pre-selected based on their colours and magnitudes, and then their distance can be determined via their period-luminosity relation. Other commonly used halo probes are blue horizontal branch (BHB) stars (e.g. Xue et al. 2008, 2011; Deason et al. 2012, 2017) , which can be selected through colour and magnitude cuts and can act as standard candles. While not exactly standard candles themselves, main-sequence turn off (MSTO) stars can also provide a distance range estimate and have been used as halo tracers (e.g. Bell et al. 2008, 2010; Kafle et al. 2017). K-Giant stars are also a typical way to map the Milky Way halo (e.g. Bond 1980; Morrison et al. 1990, 2000; Starkenburg et al. 2009; Xue et al. 2014; Deason et al. 2017). Although not standard candles, their distance can be determined (with approximately 20% uncertainty) from measurements of their astrophysical parameters (surface gravity, metallicity, and temperature) obtained from their spectra. These stars are particularly valuable kinematic tracers of the Milky Way's halo due to their high intrinsic luminosity.

Recently, Helmi et al. (2017) combined *Gaia* DR1 data with the ground-based spectroscopic survey RAVE (DR5; Kunder et al. 2017), and selected a local sample of halo stars based on their metallicity, which was further cleaned from disk contaminants by simple kinematic fits. These authors were then able to use that sample to constrain the overall degree of substructure of the halo in the solar neighbourhood. Inspired by their work, and the upcoming *Gaia* Data Release 2 (DR2), in this paper we present a supervised method for selecting halo stars. The method can be entirely data driven, and uses the position, velocity, photometry, and if available the metallicity of the stars to make the classification. This exploits the fact that halo stars have distinctly different kinematics compared to disk stars. In addition, it is expected that the stellar halo is more metal-poor than the other Galactic components (e.g. Searle & Zinn 1978; Chiappini et al. 2001), and thus its constituent stars are expected to have bluer colours on a HR diagram, which further improves the selection. We determine how viable our method is for selecting halo stars within the solar neighbourhood with *Gaia* DR2 data when having 5D phase-space information only (without line-of-sight velocities) and *Gaia* $G_{\rm BP}$, $G_{\rm RP}$, and G optical magnitudes.

This paper is structured as follows. In Sect. 2, we introduce the supervised classifier used for identifying halo stars. In Sect. 3, we test the viability of this algorithm to reliably select halo stars using the Gaia Universe Model Snapshot and mock catalogues resembling the currently available TGAS and the upcoming *Gaia* DR2 datasets. We then apply our model to the TGAS subset of *Gaia* DR1 data in Sect. 4 and present our conclusions in Sect. 5.

2. Methodology

2.1. Gradient boosted trees

To build a model for classifying halo stars, we use a technique called gradient boosted trees (Friedman 2001). In machine learning, boosting is an ensemble technique where new models are added in order to improve on the errors made by previous models. The models are added sequentially until no further improvement is made. Gradient boosting is an approach where the new models are created based on the residuals of prior models, which then are added together to make a final prediction. The term "gradient boosting" comes from the usage of the gradient descent algorithm, which is used to minimize an arbitrary differentiable loss function when adding new models. The models in this case are decision trees.

Gradient boosted trees is a powerful modelling technique with high predictive power. By combining many simple decision tree models, one can describe complicated and non-linear relationships amongst different features in a dataset. Since the base model is a decision tree, the final combined model can be easily interpreted. This is significantly better than some of the competing machine learning algorithms such as support vector machines and artificial neural networks, where it is difficult to understand how the classification boundary was drawn, especially when dealing with datasets that have high dimensionality. When using boosted trees, in contrast to most other algorithms, one does not need to scale the data or do feature engineering. In addition, this technique is robust to uninformative features, meaning that there is no penalty when training the model while using features that do not add any information towards the classification objective. This method also allows the use of sparse data. This can be quite useful, since *Gaia* DR2 may contain $T_{\rm eff}$ for a subset of stars, which may add information to the classification process.

Historically, the major drawback of gradient boosted trees is the long duration of the training process. The need to build many decision trees in succession makes this procedure difficult to parallelize and optimize, making it hard to scale up to datasets that have a large number of samples or features. In this paper, we use `XGBoost`[1] (Chen & Guestrin 2016), a state-of-the-art open source implementation of the gradient boosted trees technique, which mitigates the performance issues typically associated with this technique. It employs a novel tree building algorithm based on the studies by Li et al. (2007); Bekkerman et al. (2011); Tyree et al. (2011) and is able to use all available CPUs in a machine during training. `XGBoost` also supports distributed training, so one could use a computer cluster for building up a model, as

[1] https://github.com/dmlc/xgboost

well as out-of-core computing for datasets that are too large to fit into memory. This makes `XGBoost` a suitable tool for creating models using large tabular datasets such as *Gaia* DR2 and its subsequent releases.

2.2. Training the classifier

To train the `XGBoost` classifier, we first select the input data and associated features. These will be described in more detail in Sects. 3 and 4. The data is split into training and testing sets with proportions of 70−30%, respectively. The split is stratified, meaning that it preserves the fraction of halo to non-halo stars in both the training and testing sets.

`XGBoost` features various hyperparameters that define the behaviour of the classifier model. These parameters set the number of decision trees in the ensemble and their maximum depth, the number of features to consider when growing a tree, and control the process through which a tree is grown and pruned. One can create an optimal model for a given dataset by properly tuning these hyperparameters.

We use the Bayesian optimization package `bayes_opt`[2] based on Brochu et al. (2010); Snoek et al. (2012) to find the hyperparameters values that minimize the loss function of the `XGBoost` classifier. Bayesian optimization works by constructing a posterior distribution function in which every variable is normally distributed and every collection of random variables can be described by a multivariate Gaussian distribution, otherwise known as a Gaussian process. For each iteration, such a function is fitted to the known samples of the loss function, and then an exploration algorithm is employed to determine the combination of parameters to be used for sampling the loss function in the next step. This is an ideal method of optimizing a function for which the sampling is very computationally expensive, since it minimizes the number of iterations needed to find a parameter combination that is close to optimal.

At each step of the Bayesian optimization, the value of the loss function is taken to be the mean loss coming from a five fold cross-validation. This is done as follows. First, the training set is split into five subsets in a stratified manner. The model is then trained on four of those subsets, while the remaining subset is used to asses the model performance based on the logarithmic loss function. In machine learning, the logarithmic loss ($\mathcal{L}$) quantifies the accuracy of a classifier by penalizing false classifications, and in the case of only two classes it is defined as

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1-y_i)\log(1-p_i)], \tag{1}$$

where N is the number of samples, y_i is a binary indicator of whether a sample has been correctly identified, and p_i is the probability of assigning a positive label to that sample. This is repeated five times and each time a different one of the five subsets is used to evaluate the model. Throughout this process we also monitor the precision, recall, and the Matthews correlation coefficient. In the following definitions of these metrics, TP stands for "true positives", that is, stars that were correctly labelled as halo, TN stands for "true negatives" or stars that were correctly identified as non-halo, FP or "false positives" are stars wrongly assigned as halo, and FN or "false negatives" are halo stars what have been mislabelled as non-halo. With this in mind, the recall can be expressed as

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{2}$$

and measures the fraction of the positive class that is being recovered, and in this case that is the fraction of correctly identified halo stars. A value of 1 means that all halo stars were successfully recovered. The precision is defined as

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{3}$$

which measures the level of contamination present in the sample of positive predictions. A precision value of 0.8 means that there is 20% contamination in the predicted sample of halo stars.

The Matthews correlation coefficient is designed to measure the performance of a binary classifier and it takes into account both the false positive and false negative predictions, and it is defined as.

$$\text{MCC} = \frac{\text{TP}\times\text{TN} - \text{FP}\times\text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}}. \tag{4}$$

A value of 1 signifies a perfect model, 0 represents random guessing while a value of −1 means a total disagreement between the features and the classification.

Monitoring these metrics simultaneously is very important when trying to asses a model, which is designed to predict the minority class of a highly imbalanced dataset. For the final several steps of the Bayesian optimization procedure, we check that the standard deviations of the logarithmic loss, precision, recall, and the Matthews correlation coefficient are small. The small deviation of these statistics tells us that the model performance is stable when the training and the testing data is varied, meaning that it does not suffer from over-fitting. In general we find the `XGBoost` classifier to perform quite well for a sensible choice of parameters, and we observe only a slight improvement in the results after undergoing the rigorous tuning described above.

Once the hyperparameters of the classifier are tuned and we are satisfied with the performance of the model judging from the cross-validation scores (recall, precision, Matthews correlation coefficient) we proceed to evaluate the predictive power of the model via the so-far unused test set. In Sect. 3, we present and discuss the results obtained from this final evaluation of the model.

3. Identifying halo stars in GUMS

To test the viability of our classifier to detect halo stars, we first apply it to data extracted from the *Gaia* Universe Model Snapshot (GUMS; Robin et al. 2012), based on the Besançon Galaxy Model (Robin et al. 2003). Due to how GUMS is generated, the stellar sources are labelled as belonging to one of the four galactic components, thin disk, thick disk, bulge, and halo. Here, we focus on how well we can identify halo stars in TGAS-like and *Gaia* DR2-like data. To create a TGAS-like set, we produce a magnitude limited sample that is used to evaluate the best possible performance of the model in the solar neighbourhood. This is an ideal scenario because the data is perfect as it contains no measurement or systematic uncertainties. We then convolve this sample with the median uncertainties from the TGAS data, in order to evaluate the performance of our model

[2] https://github.com/fmfn/BayesianOptimization

under more realistic conditions. We also select a much larger sample from GUMS, which we error convolve with the expected uncertainties from *Gaia* DR2.

3.1. The ideal solar neighbourhood sample

The stars in the TGAS-like sample extracted from GUMS have magnitudes between $6 \leq G \leq 12.5$, but also $0.2 \leq \log(g) \leq 5$, $3000 \leq T_{\rm eff} \leq 9000$ K. The magnitude criterion is chosen such that this sample loosely resembles the TGAS subset of *Gaia* DR1 (Gaia Collaboration 2016; Lindegren et al. 2016), while the cuts on $T_{\rm eff}$ and log(g) are chosen considering the RAVE DR5 data. Our TGAS-like sample contains ∼4.7 million stars, of which 9240 (2%) belong to the halo component according to the labels in GUMS.

Figure 1 shows the distance and [Fe/H] distributions for these stars. From the distance distribution, one can see that the halo number count is roughly constant out to a distance of ∼8 kpc, and that disk stars significantly outnumber the nearby (<2 kpc) halo. On the other hand, the halo stars are systematically more metal-poor than the rest, as shown in the bottom panel in Fig. 1.

On the left panel in Fig. 2, we show the velocity distribution in Cartesian coordinates of our GUMS sample, where the blue points mark the halo stars. One can see that the halo stars have distinctly different kinematics than the rest. The disk stars rotate around the Galactic centre with a mean $v_y \sim 200$ km s^{-1}, while the stellar halo shows no such orderly motion, and the means of its velocity components are centred close to zero. The right panel on the same figure shows a HR diagram. The halo stars show systematically bluer colours compared to the rest, in line with their [Fe/H] distribution shown in Fig. 1. It is this combination of their characteristic kinematics, bluer colours, and low [Fe/H] values that we rely on to train the classifier to separate the halo stars from the rest.

To asses the ability of our classifier to identify halo stars, we consider three cases depending on the data that may be available for the training process. In the first case, we use the full phase-space information, together with the absolute G magnitude (M_G), the optical $G_{\rm BP}-G_{\rm RP}$ colour, and the [Fe/H] for all stars in our GUMS sets. This is the best case scenario in terms of data availability. It is unlikely that [Fe/H] estimates will be released as part of *Gaia* DR2, however we find it useful to test the performance of our classifier under optimal conditions. In the second test case, which is the same as above but without metallicity information, we use features that we know will be available in *Gaia* DR2, at least down to magnitude $G \sim 12.5$: 3D positions and velocities, coupled with *Gaia* photometry. These quantities are expected to be available for ∼5 million stars in *Gaia* DR2. Our third scenario is designed to be applicable to the currently available TGAS dataset: we use the 3D positions, proper motions, and *Gaia* photometry of the stars, without any knowledge of their line-of-sight velocities. In this case, instead of transforming the proper motion and line-of-sight velocities according to the equations

$$v_x = v_{\rm los}\cos(l)\cos(b) - k\,d\,\mu_l\sin(l) - k\,d\,\mu_b\cos(l)\sin(b), \quad (5)$$

$$v_y = v_{\rm los}\sin(l)\cos(b) + k\,d\,\mu_l\cos(l) - k\,d\,\mu_b\sin(l)\sin(b) + v_{\rm LSR}, \quad (6)$$

$$v_z = v_{\rm los}\sin(b) + k\,d\,\mu_b\cos(b), \quad (7)$$

we assume $v_{\rm los} = 0$ km s^{-1} and derive pseudo-Cartesian coordinates $(\tilde{v}_x, \tilde{v}_y, \tilde{v}_z)$. In the above equations, l and b are the longitude and latitude of the stars in Galactic coordinates, while μ_l and μ_b are the associated proper motions. The distance is denoted by d in units of kpc measured from the Sun, $v_{\rm los}$ are the line-of-sight velocities of the stars in units of km s^{-1}, while $k = 4.74057$ is a scaling factor that puts the velocities (v_x, v_y, v_z) in units of km s^{-1}. In Eq. (6), we correct the v_y velocity for the motion of the local standard of rest $v_{\rm LSR}$, which we assume to be 220 km s^{-1}. In the middle panel of Fig. 2, we show the distribution of all stars in our solar neighbourhood sample in the $\tilde{v}_x$–$\tilde{v}_y$ space. The halo stars that have $v_{\rm los} \sim 0$ km s^{-1} are very weakly affected by this transformation. This is also true for the majority of the disk stars in the solar neighbourhood, which are known to have $v_{\rm los} \sim 0$. On the other hand, the halo stars with larger $v_{\rm los}$ but small tangential motions are seen to cluster around $\tilde{v}_y \sim 200$ km s^{-1}. In general, even with this transformation, one can see that there is a clear distinction between the overall kinematics of the halo stars compared to the vast majority of stars belonging to the other Galactic components.

For each of the three cases described above, we train a gradient boosted trees classifier as described in Sect. 2.2. The results of the evaluation of these three cases are summarized in Table 1 according to the metrics we follow during the training

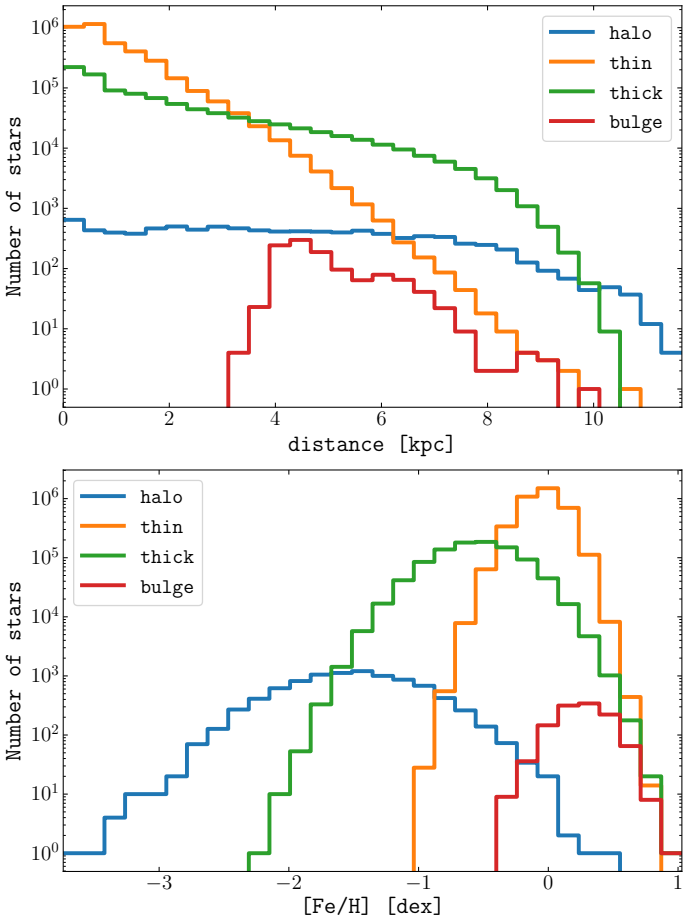

Fig. 1. Distance (*top panel*) and [Fe/H] (*bottom panel*) distributions for each of the Galactic components of the stars in the solar neighbourhood from the perfect GUMS sample. There are ∼4784319 stars in this dataset, 9240 of which are halo. The halo stars have a roughly constant distribution out to a distance of ∼8 kpc from the Sun. While they span a large range of [Fe/H] value, the bottom panel shows that the halo stars are on average more metal-poor than stars belonging to the other Galactic components.

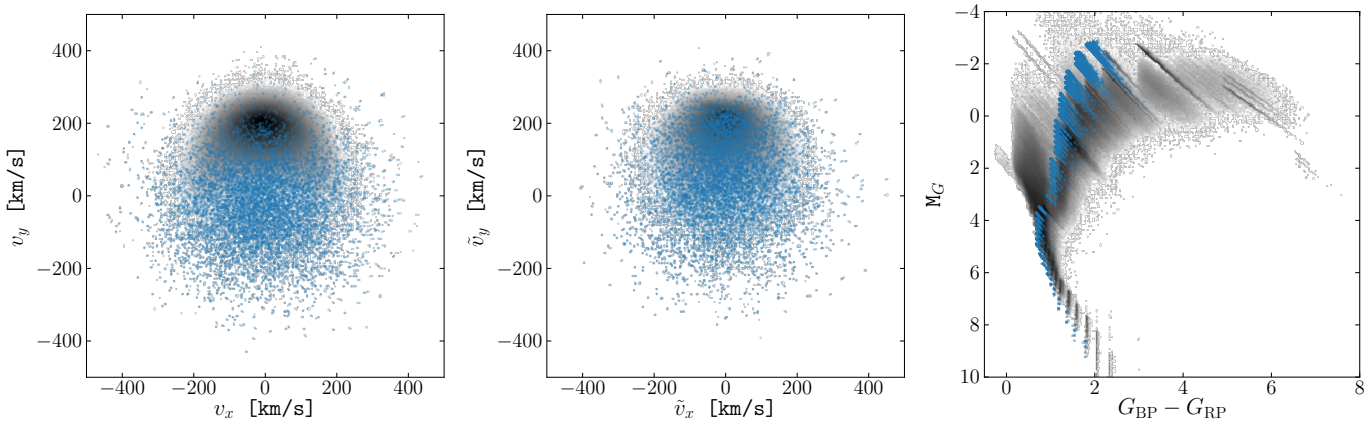

Fig. 2. *Left panel*: Cartesian velocities of the GUMS sample in the solar neighbourhood. The blue points mark the halo stars, while the underlying density distribution shows all the stars in the sample. One can readily notice that the halo members have distinctly different kinematics compared to the rest of the stars. *Middle panel*: equivalent velocity distribution calculated assuming $v_{\rm los} = 0$ km s^{-1}, for the case when the radial velocities of the stars are not known. The halo stars with $v_{\rm los}$ close to 0 are not affected by this transformation, while the stars with large $v_{\rm los}$values cluster around $\tilde{v}_y \sim 200$ km s^{-1}. There is still a kinematic distinction between the halo stars and the rest. *Right panel*: HR diagram, where the symbols are the same as on the *left panel*. The halo stars have systematically bluer colours, in line with their [Fe/H] distribution.

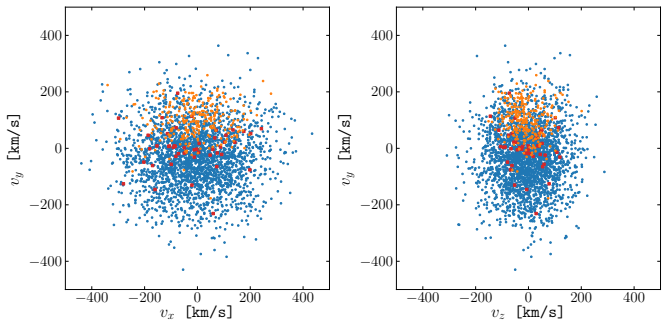

Fig. 3. Velocity distribution in the v_x–v_y space (*left panel*) and in the v_z–v_y space (*right panel*) of the stars labelled as halo (true positives, blue points), stars wrongly labelled as halo (false positives, red squares), and stars wrongly labelled as non-halo (false negatives, orange points), for the test case when using photometry and full phase-space information on the ideal GUMS selected sample in the Solar neighbourhood.

and optimization process. From this table, one can see that in the cases when we use full phase-space information of the stars as a training feature, we detect nearly all the halo stars in the test sample. In addition, the level of contamination is negligible. Knowledge of the metallicity is not necessary, as case two shows, especially when there are no measurement uncertainties, the $G_{\rm BP}$–$G_{\rm RP}$ colour is an excellent proxy for [Fe/H] for the halo stars. Figure 3 shows the velocity distribution, corrected for the LSR, for the correctly identified halo stars (blue), false positives (red), and false negatives (orange). It can be seen that the stars which were incorrectly labelled as non-halo have kinematics similar to the disk.

In the final test case, when we do not know the $v_{\rm los}$, we recover over 84% of the halo stars. The precision is 98%, which means that the degree of contamination is negligible. The left panel in Fig. 4 shows the velocity distributions of the recovered halo stars, as well as that of the false positives and the halo stars missed by the classifier (false negatives). It is interesting to see that the majority of the false negatives have $\tilde{v}_y \sim 200$ km s^{-1}, and are in fact those stars for which the conversion to the Cartesian velocity coordinate system is least reliable, that is, for the stars that had larger line-of-sight velocities and small tangential motions. The right panel in Fig. 4 shows a HR diagram, where one can see that the classification is most reliable for stars with very blue $G_{\rm BP}$–$G_{\rm RP}$ colours, especially if they are giants. Moreover, we see no difference in the metallicity distribution of the correctly classified halo stars and stars falsely labelled as non-halo. This mislabelling is mainly due to not knowing their line-of-sight velocities.

3.2. Detecting halo stars in errors-convolved GUMS samples

The excellent detection rates we report in Sect. 3.1 are largely due to the dataset being ideal. Since our goal is to identify halo stars in the TGAS and the upcoming *Gaia* DR2 datasets, we test the performance of our classifier on two mock catalogues that resemble these datasets.

3.2.1. TGAS uncertainties

We create a mock catalogue, which resembles TGAS, by convolving the observables in the GUMS sample selected above with the median uncertainties from TGAS listed in Table 2. Since TGAS does not provide $v_{\rm los}$ and [Fe/H], for these we took the median uncertainties from the cross-match between TGAS and RAVE. Since this mock catalogue is not ideal but contains measurement uncertainties, computing the distance to stars by inverting the associated parallax may not be reliable in all cases and can lead to biases in the data. Assuming that the parallax uncertainty distribution is Gaussian, Helmi et al. (2017) have shown that for relative parallax error $\leq$30%, the distance computed by taking the reciprocal values of the parallax is not too biased compared to the true distance. Therefore, we proceed to work only with stars that have relative parallax error $\leq$30%. Such a selection leaves ~3 million stars in the GUMS error convolved sample with TGAS uncertainties, of which only 1379 are halo according to the data model (~0.046% of the sample).

As before, we train the halo star classifier as described in Sect. 2.2. The choice to both train and evaluate the classifier on the error convolved data makes for a more generalized model. Training the model on the ideal and applying it on the error convolved set may cause the classifier to under-perform, since the two sets, training and testing, would be drawn from different distributions.

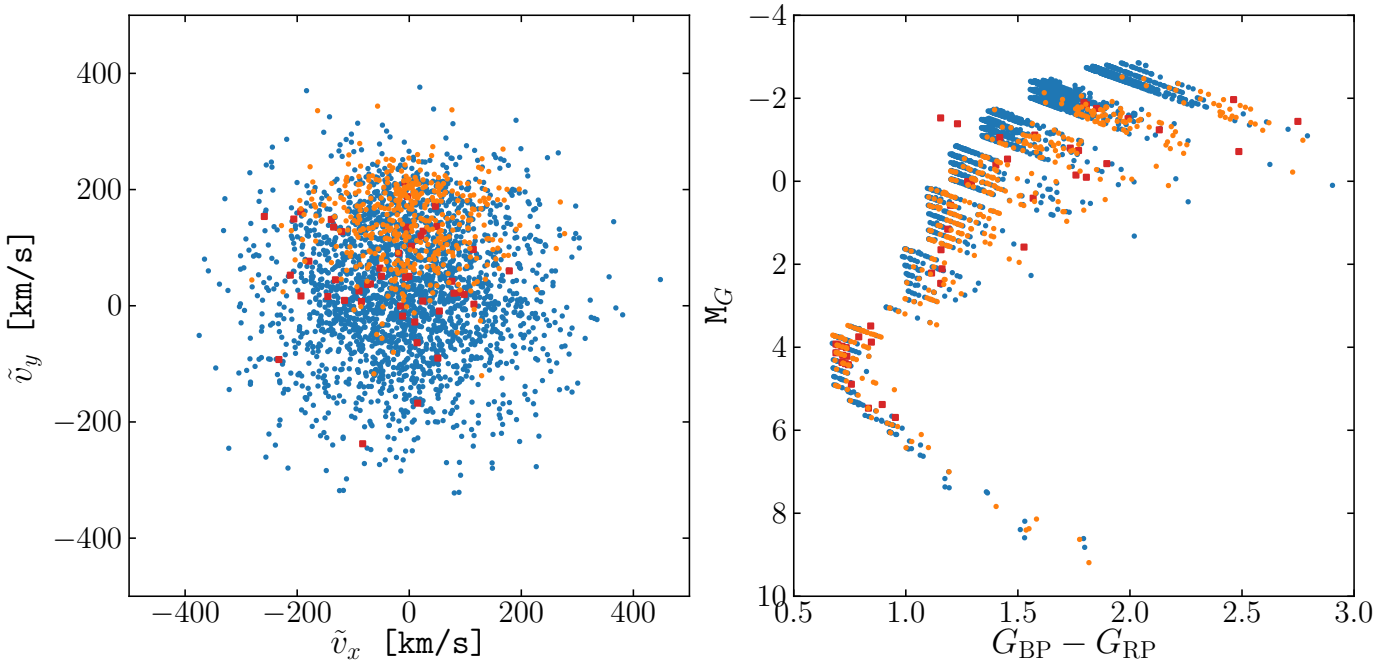

Fig. 4. Diagnostics for the model trained on the ideal GUMS solar neighbourhood sample, using only the *Gaia* optical photometry and 5D phase-space information. *Left panel*: velocity distribution in Cartesian coordinates, assuming $v_{\rm los} = 0$ km s^{-1}. *Right panel*: HR diagram. The blue points mark the true positive (correctly identified halo stars), the orange points mark the false negatives, while the red squares mark the false positives. Not knowing the line-of-sight velocities of the test stars makes for an increase in the numbers of false negative detections.

Table 1. Metrics describing the performance of our halo star classifier for the three different cases depending on the training features available.

	Ideal GUMS sample with $G < 12.5$ mag		
	6D phase-space + phot +[Fe/H]	6D phase-space + phot	5D phase-space + phot
recall	0.94	0.90	0.84
precision	0.99	0.98	0.98
logarithmic loss	0.01	0.01	0.01
Matthews coef	0.96	0.94	0.90
	GUMS sample with $G < 12.5$ mag convolved with median TGAS uncertainties		
	6D phase-space + phot +[Fe/H]	6D phase-space + phot	5D phase-space + phot
recall	0.71	0.64	0.46
precision	0.90	0.87	0.82
logarithmic loss	0.01	0.01	0.01
Matthews coef	0.80	0.75	0.62
	Error convolved GUMS sample with *Gaia* DR2 uncertainties		
	6D phase-space + phot +[Fe/H][a]	6D phase-space + phot[a]	5D phase-space + phot[b]
recall	0.93	0.90	0.58
precision	0.98	0.98	0.92
logarithmic loss	0.01	0.01	0.02
Matthews coef	0.95	0.94	0.73

Notes. The upper bound of the uncertainty of these metrics, calculated as the standard deviation from the five fold cross-validation process during the training, is 0.03. The superscripts *a* and *b* indicate that the model was evaluated on GUMS selected samples with $G < 12.5$ mag, and $G < 19$ mag respectively.

The performance of the classifier is shown in Table 1. One can see that after adding TGAS-like uncertainties, the performance of the classifier does indeed decline. In the worst case scenario, in which we do not have line-of-sight velocity information, the classifier is able to recover only 46% of the halo stars, with a contamination level of 18%. In fact, given the magnitude of the uncertainties in the observables, the precision of the model has remained exceptionally high, with the contamination level never reaching 20%. This means that the halo candidates selected by the classifier have a high probability of being true halo stars, even after the error convolution.

3.2.2. *Gaia* DR2 uncertainties

The primary reason for building this classifier is to detect halo stars in *Gaia* DR2. To test the ability of the classifier to do this, we create a second mock catalogue that roughly resembles *Gaia* DR2 by selecting stars in GUMS that have $G \leq 19$, $0.2 \leq \log(g) \leq 5$, and $3000 \leq T_{\rm eff} \leq 9000$ K. The sample is then convolved with the expected uncertainties for *Gaia* DR2 according to the following relations for the astrometry:

$$\begin{aligned}
\sigma_\pi[\mu\text{as}] &= 0.9965\,(-1.631 + 680.766\,z_G + 32.732\,z_G^2)^{0.5}\,t_{\rm frac}^{0.5} \\
\sigma_\alpha[\mu\text{as}] &= 0.787\,\sigma_\pi \\
\sigma_\delta[\mu\text{as}] &= 0.699\,\sigma_\pi \\
\sigma_{\mu_\alpha}[\mu\text{as yr}^{-1}] &= 0.556\,\sigma_\pi\,t_{\rm frac} \\
\sigma_{\mu_\delta}[\mu\text{as yr}^{-1}] &= 0.496\,\sigma_\pi\,t_{\rm frac},
\end{aligned} \tag{8}$$

where $z_G = \text{MAX}[10^{0.4\,(12.09-15)}, 10^{0.4\,(G-15)}]$ and $t_{\rm frac} = 60/22$ is the total number of months for which the nominal *Gaia* mission is scheduled to run over the number of months during which the DR2 data is observed (A. G. A Brown, priv. comm.). For the photometry we assume

$$\begin{aligned}
\sigma_G[\text{mag}] &= 10^{-3}\,(0.04895\,z_G^2 + 1.8633\,z_G + 0.0001985)^{0.5} \\
\sigma_{\rm BP/RP}[\text{mag}] &= 10^{-3}\,(10^{a_{BP/RP}}\,z_G^2 + 10^{b_{BP/RP}}\,z_G + 10^{c_{BP/RP}})^{0.5},
\end{aligned} \tag{9}$$

where $(a_{BP}, b_{BP}, c_{BP}) = (1.334, 1.623, -1.987)$, and $(a_{RP}, b_{RP}, c_{RP}) = (1.199, 1.576, -3.096)$.

We assume that the uncertainties of all observables are Gaussian in nature. For comparison, Table 2 lists the expected median uncertainties for a star with $G = 17$ mag. This clearly shows the major improvement in *Gaia* DR2, where even for a much fainter star, the astrometric and photometric uncertainties of *Gaia* DR2 are superior to those of the brighter TGAS sample. For reference, a star with $G = 12.5$ in *Gaia* DR2 is expected to have a median astrometric uncertainty of 0.02 mas, and its proper motion uncertainty is expected to be 0.03 mas/yr.

This mock catalogue contains ∼620 million stars in total. Following the error convolution, we again discard stars that have negative parallaxes or relative parallax uncertainties larger than 30%. The resulting sample contains ∼200 million sources of which only ∼0.25% belong to the halo component. Even though this cut decreases the total number of stars by a factor of three, it removes 99% of the halo stars.

The *Gaia* DR2 catalogue will contain full phase-space information only for stars brighter than $G \sim 12.5$. Thus, in the test cases in which we use stars that have full phase-space information, we only consider stars that have magnitudes brighter than $G = 12.5$. Even though the second data release of *Gaia* will not feature metallicity estimates, in our first test case we do use [Fe/H] together with the astrometric, velocity, and photometric data during the training and evaluation in order to gauge the optimal performance of the classifier. In the second scenario, we use the features we know are going be available in the actual data release in this bright magnitude range. These first two scenarios are equivalent to those when we used the GUMS sample in the solar neighbourhood convolved with TGAS errors, but now the GUMS data is convolved with the expected errors for *Gaia* DR2.

In the third test case, we use our entire *Gaia* DR2-like mock catalogue to identify potential halo stars, using only the 5D astrometric solution and the photometric data. Since the stars in this entire mock catalogue are no longer concentrated in the solar neighbourhood but span a much larger volume, we convert their positions and velocities to a Galactic cylindrical coordinate system $(R, \phi, z, \tilde{v}_R, \tilde{v}_\phi, \tilde{v}_z)$. For the velocity transformations, we assume the line-of-sight velocity of each star to be 0 km s^{-1}. From the 200 million sources that comprise this mock catalogue, we use 150 million for the training process, and the rest to evaluate the model. During the splitting of the data, we made sure there are equal proportions of stars in a given magnitude range to the total number of stars in both the training and the test set.

The performance of the classifier for each of the three scenarios outlined above is listed in Table 1. In the cases when full phase-space information is available, the classifier recovers at least 90% of the halo stars in the test set, with less than 2% contamination. In fact, its performance is nearly equally as good as when trained and applied on the ideal, error-less data. This is mainly due to the high precision measurements expected for the *Gaia* DR2 data.

In the final test case, in which we do not know the line-of-sight velocities of the stars, our classifier recovers nearly 60% of the stars in the test set, with less than 10% contamination. The top panel on Fig. 5 shows the on-sky distribution of the correctly classified halo stars (blue points), the false positives (red points), and the false negatives (orange points) for this case. One can see that the stars correctly classified as halo are uniformly distributed on the sky, while the stars wrongly labelled as halo or non-halo, tend to inhabit the regions closer to the Galactic plane ($|b| < 25$ deg).

The bottom left panel in the same figure shows the velocity distribution of the stars in the Galactocentric cylindrical coordinate system. One can see that the majority of the stars wrongly classified as non-halo are centred at $\tilde{v}_\phi \sim 180$ km s^{-1}, and have kinematics similar to those of disk stars. The bottom middle panel shows a HR diagram where the wrongly labelled non-halo stars are seen to have systematically redder colour, or to be systematically fainter than the correctly recovered halo stars. From this we conclude that our classifier only has problems in identifying halo stars that have kinematics similar to those of disk stars, while at the same time having redder $G_{\rm BP}$–$G_{\rm RP}$ colours.

The bottom right panel in Fig. 5 shows distance distributions from the Sun for the stars correctly classified as halo (blue histogram), stars wrongly classified as non-halo (orange histogram), and stars falsely labelled as halo (red histogram).

The top and bottom left panels in Fig. 6 display the performance of our classifier as a function of Galactic (l, b) coordinates according to the three key diagnostic metrics (recall, precision, and the Mathews correlation coefficient) we use to evaluate our final test case. One can see that the model detects halo stars rather well across the sky, with the exception of the region spanned by the Galactic disk where we see an expected drop in performance, particularly in the recall and the Mathews correlation coefficient. On the other hand, the precision of the model is rather uniform across the entire sky, meaning that our halo sample will have a negligible level of contamination even in the direction of the disk. In the bottom right panel of the same figure, we show how the classifier performs as a function of G magnitude. The panel shows that the model is most effective for the bright nearby sample of stars. The precision again remains exceptionally high for any magnitude bin.

3.3. Comparison to the classical kNN algorithm

As an additional test on the suitability of our boosted trees model to identify halo stars, it is useful to compare its performance to other classification algorithms or methods. In this section,

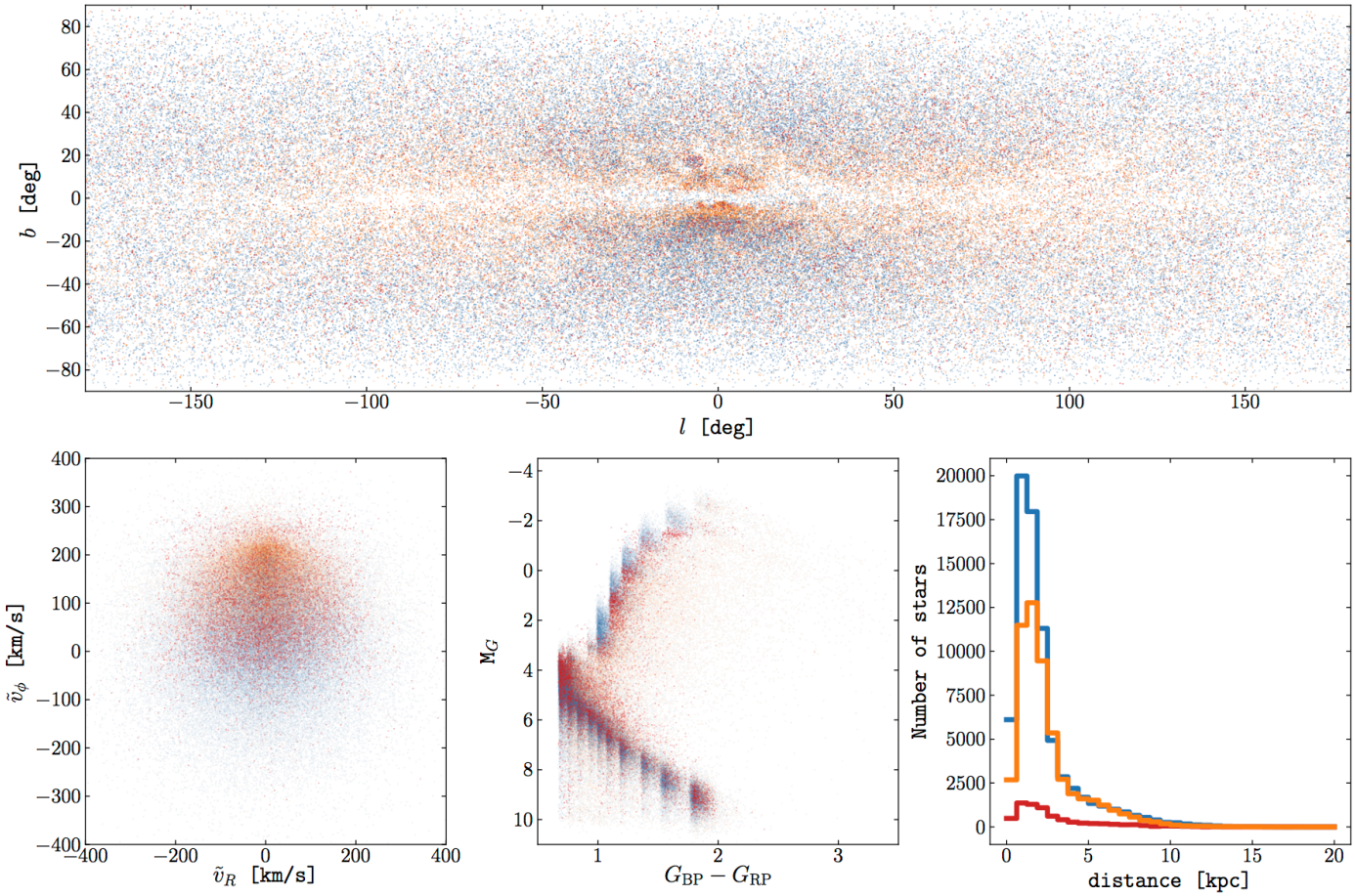

Fig. 5. Performance of the classifier when applied to the error convolved GUMS data with *Gaia* DR2 uncertainties, in the case when only the 5D phase-space coordinates and the optical photometry of the stars are used as training features. *Top panel*: sky distributions of the correctly detected halo stars (blue), false negatives (orange), and false positives (red). *Bottom left panel*: velocity distribution in the Galactic cylindrical pseudo velocity plane $\hat{v}_\phi$ vs. $\hat{v}_R$ ($v_{los} = 0$ km s^{-1}). *Bottom middle panel*: HR diagram of the stars shown on the top panel. Symbols are as on the top panel. *Bottom right panel*: distance distribution of the stars in the sample. The histogram colours match the colours of the points in the other panels.

Table 2. Median standard deviations of the uncertainty distributions of the observables with which we convolve the ideal GUMS selected data to create more realistic catalogues.

Observable	σ_{TGAS}	$\sigma_{DR2}(G = 17)$	unit
RA	0.23	0.10	(mas)
Dec	0.21	0.09	(mas)
parallax	0.32	0.12	(mas)
μ_{RA}	0.99	0.19	(mas yr^{-1})
μ_{Dec}	0.83	0.17	(mas yr^{-1})
v_{los}	1.08	2.50	(km s^{-1})
G	3×10^{-2}	9×10^{-3}	(mag)
G_{BP}	3×10^{-2}	8×10^{-2}	(mag)
G_{RP}	3×10^{-2}	7×10^{-2}	(mag)
[Fe/H]	0.25	0.25	(dex)

Notes. In the case of *Gaia* DR2, the uncertainties are those for a star with $G = 17$.

we compare the performance of our model to that given by the classical k-Nearest-Neighbours (kNN) algorithm.

The kNN (e.g. Cover & Hart 2006) is one of the classical machine learning algorithms for classification and pattern recognition. In its most basic form the algorithm is quite simple. The training phase only stores the feature vectors and the classification labels of the training data. A new data point is then labelled to have the most frequent class of its nearest k neighbours (hence the name) in the space spanned by the feature vectors being considered. It is quite common to use the Euclidean distance. One could extend the algorithm by weighting the distances between the data we are trying to predict and its k nearest neighbours, give more weight to certain features, or consider different distance metrics, for example. Here, we use the kNN method in its simplest form, as implemented in the `scikit-learn` library.

We proceed to apply the kNN algorithm to our mock data with *Gaia* DR2 uncertainties and we use the same training features subject to the same quality cuts as in the case of the boosted trees model. Prior to applying the algorithm, we scale each feature of the training and test data to have zero mean and unit variance, which is standard procedure when using the kNN algorithm with a Euclidean distance metric. Then we run the algorithm through our training dataset with different values for k, the number of neighbours to be considered for the classification process. We found $k = 31$ to give the optimal performance. Applying this method to the bright mock sample with *Gaia* DR2 errors, for which we have full phase space information, the kNN model achieves a recall of 0.59 and a precision of 0.96. When applied to the full *Gaia* DR2 mock sample, with 5D phase-space information, the kNN classifier reaches a recall of 0.42 and a precision

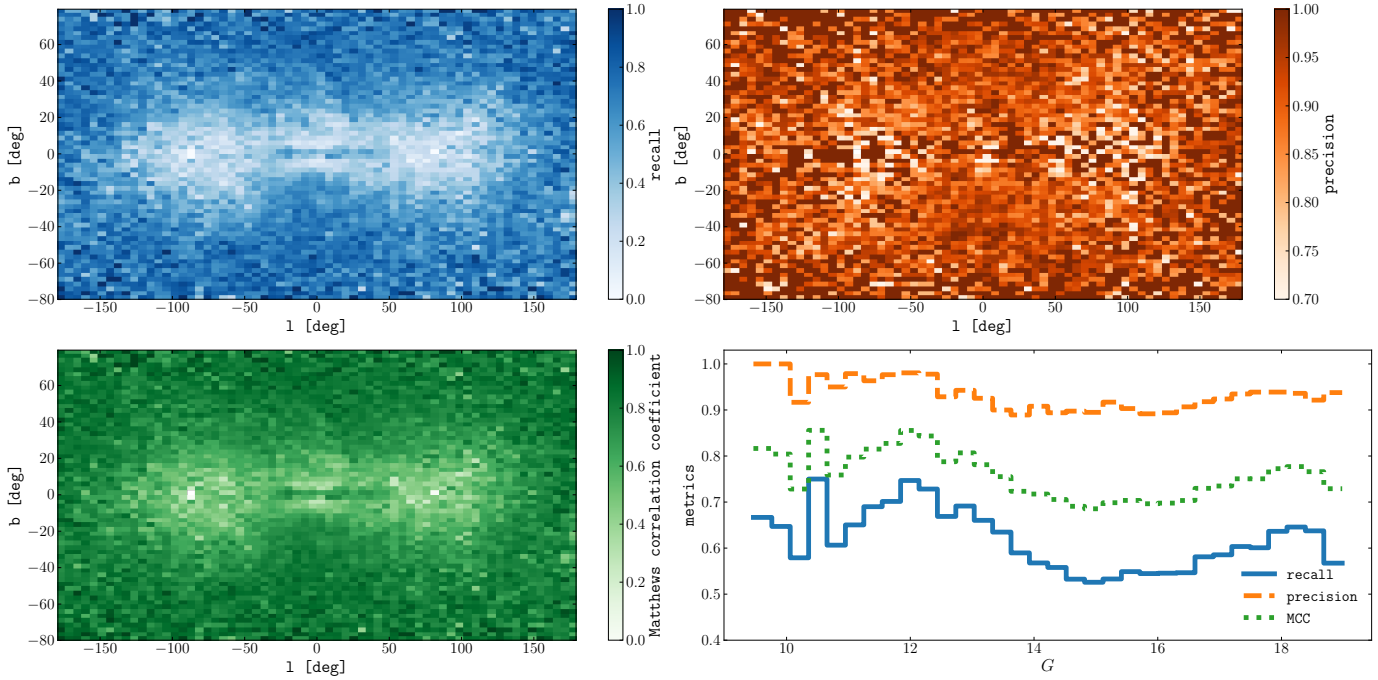

Fig. 6. Performance of the classifier as a function of Galactic (l, b) coordinates and the apparent magnitude G for each of the three key diagnostic metrics, when applied *Gaia* DR2-like data (without line-of-sight velocity information, i.e. the dataset used also in Fig 5.

of 0.92. For this test, we see that, while the *kNN* performs admirably well, the boosted trees model is superior in detecting a larger fraction of halo stars. This is not surprising, as the *kNN* method is typically used as a clustering algorithm, while the halo stars are not clustered in the feature space we considered. In addition, *kNN* does not scale well for large datasets such as our *Gaia* DR2 sample, since in order to assign a class label to each new data point, it needs to calculate the distance to all samples in the training set. For the dataset used in this work, the *kNN* algorithm is ∼4 times slower compared to the `XGBoost` model.

4. Detecting halo stars in TGAS

The exercises we described in the Sect. 3 are entirely model dependent: the definition of what constitutes a halo star is completely defined by the GUMS data model. They are meant as a proof of concept, aiming to show the capabilities of our halo classifier given the confines of the data model.

In this section, we apply our halo classifier to the entire TGAS dataset. In what follows we describe the creation of our training set and how we assign the halo labels. Later in Sect. 4.2 we apply the trained classifier to the TGAS dataset. The approach we present here is fully model independent, in the sense that we use an entirely data-driven approach to define the halo labels using an extended number of stellar features, and then train the classifier to be able to detect halo stars based on a limited feature set.

4.1. Defining the training sample

In order to detect halo stars in an entirely model-independent manner, we follow the example set by Helmi et al. (2017), which combined metallicity and kinematics criteria exploiting the synergy between TGAS and RAVE. For a more detailed discussion on this selection process, we refer the interested reader to Sect. 2 in Helmi et al. (2017). Here, we provide a summary of the selection criteria and steps to label stars as halo. In this work, we use the TGAS dataset cross-matched to RAVE from McMillan et al. (2018), which used the TGAS parallaxes as priors to derive the spectrophotometric parallaxes.

We only consider stars that have a velocity uncertainty $\Delta_{RV} \leq 10$ km s^{-1}, CorrCoeff ≥ 10, $S/N \geq 20$, and algoConv $\neq 1$. These criteria ensure that the stars in the RAVE data have reliable radial velocities and astrophysical parameters. In addition to this, we again only select stars that have relative parallaxes with an uncertainty better than 30%. We use the updated spectrophotometric parallaxes from RAVE whenever they have better relative uncertainties compared to the parallaxes from TGAS.

In order to label halo stars after imposing the above quality selection criteria, as in Helmi et al. (2017), we select stars that have $[M/H] \leq -1$ and distances greater than 0.1 kpc. In this work, we use the metallicity estimates provided by McMillan et al. (2018), which are calibrated to match external catalogues, and this is why the metallicity threshold is different here compared to Helmi et al. (2017).

The improved log(g), $T_{\rm eff}$, and $[M/H]$ by McMillan et al. (2018) lead to a low contamination level by disk stars in the metal-poor halo candidates as judged by looking at their velocities. Still, some disk contamination remains and we model this by fitting a two-component Gaussian mixture model to the Cartesian velocity coordinates (v_x, v_y, v_z). One of the Gaussians is centred at $v_y \sim 0$ km s^{-1}, and the stars that have a higher probability of being drawn from this component are labelled as halo stars in our training set. The other component is centred on $v_y \sim 180$ km s^{-1}, and the stars that have a higher probability of being drawn from this Gaussian are eliminated as disk contaminants. The mean velocity of the disk contaminants in this case seems to be lower than the typically assumed velocity of the local standard of rest, most likely because our low metallicity sample is mostly contaminated by thick disk stars. With this approach we label 1217 stars in total to be halo.

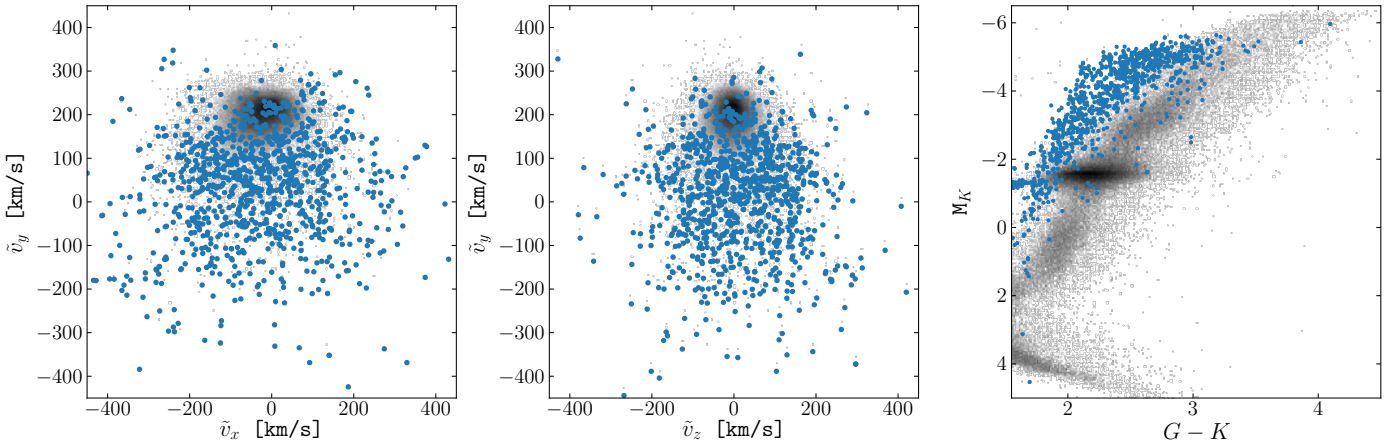

Fig. 7. TGAS × RAVE data used to train the TGAS halo star classifier. The blue points mark the locations of high confidence halo stars selected on the basis of their metallicities and kinematics, in an entirely model-independent way. The underlying density map shows the remaining stars after imposing our quality criteria on a logarithmic scale (see Sect. 2.2 and Helmi et al. 2017 for details).

Figure 7 shows the training set based on the cross-match between TGAS and RAVE. The first two panels show the location of the halo stars (blue points) in the Cartesian velocity space. Although the Gaussian velocity decomposition was done in (v_x, v_y, v_z), here we show $(\tilde{v}_x, \tilde{v}_y, \tilde{v}_z)$ since these features are used in the training process. The rightmost panel shows the location of the halo stars in a HR diagram, for which we used 2MASS photometry (Skrutskie et al. 2006). The entire TGAS dataset overlaps with the 2MASS catalogue, and since *Gaia* DR1 released photometry only in the broad G band, we rely on the 2MASS photometry for the construction of HR diagrams when dealing with the TGAS data.

Since nearly all of the halo stars identified in the TGAS × RAVE data are located on the red giant branch, there are not enough main sequence halo stars to reliably fit the model. Thus, to lower contamination, we introduce a $G-K > 1.55$ colour cut in the training set. Finally, our complete TGAS × RAVE training set comprises 119728 stars, of which 999 are labelled as halo.

4.2. Classifying halo stars in TGAS

We train our halo star classifier on the TGAS × RAVE training set following the same procedure as described in Sect. 2.2. To train the model we use the positions of the stars (x, y, z), the velocities $(\tilde{v}_x, \tilde{v}_y, \tilde{v}_z)$, the $G-K$ colour, and the absolute magnitude M_K. After optimizing and training the classifier, we obtain a precision of 0.90 ± 0.03 and a recall of 0.63 ± 0.04 on a fully unseen test sample, which comprises 30% of the complete training set. The statistics from this training process are better than those for the GUMS sample of stars selected in the solar neighbourhood convolved with TGAS uncertainties. This is because in this case we are using the updated spectrophotometric parallaxes from McMillan et al. (2018), which have smaller uncertainties compared to the TGAS parallaxes.

Prior to feeding the TGAS dataset into the halo classifier, we apply certain selection criteria assuring that we are using appropriately high quality data. In a similar manner as when constructing the training dataset, we eliminate all stars in TGAS that have relative parallax uncertainties larger than 30%. In addition, to lower contamination from nearby disk dwarfs, we exclude all stars with distances smaller than 0.1 kpc. This leaves us with 423431 stars.

Feeding this dataset into our halo classifier, we detect 337 highly probable halo candidate stars. Figure 8 shows the on-sky and velocity distributions of the halo star candidates (blue points), as well as their location on a HR diagram. From this figure, one can see that all stars have kinematics consistent with being halo and are located in the region of the HR magnitude diagram typically associated with metal-poor stars. From the top panel on Fig. 8, we see that the identified halo stars are not distributed uniformly on the sky. This is mainly due to the training set not covering the entire sky. In addition, since the training set is partially a sub-sample of the input TGAS data, we find 113 halo stars in common between the supervised classification of TGAS and the unsupervised classification based on TGAS × RAVE.

At first glance, detecting only 337 halo stars may seem too small a number given the input catalogue of 423431 sources, or may indicate a poor performance of the classifier. The small number of detections seems to be directly related to the parallax uncertainties in the TGAS dataset. When constructing our training sample using TGAS × RAVE, the vast majority of the labelled halo stars are more distant giants for which we used the superior spectrophotometric parallaxes from RAVE (McMillan et al. 2018). In fact, only 61 out of the 1217 halo stars we detected in TGAS × RAVE have parallaxes that come from the TGAS dataset. Of those, only 16 are giant stars, while the remaining 45 are closer dwarf stars. Furthermore, the fraction of halo stars we detect in TGAS (0.079%) is not very different from the fraction of halo stars present in our TGAS-like sample drawn from GUMS (~0.046%). In addition, the near absence of dwarf halo stars in the training set makes our model heavily biased towards the detection of red giant halo stars.

In order to asses whether the classifier is performing sensibly, we looked at the very basic kinematic properties of the selected halo sample. The mean velocity of the halo component in Cartesian coordinates is $(\tilde{v}_x, \tilde{v}_y, \tilde{v}_z) = (14, -10, 28)$ km s^{-1}, while their associated velocity dispersions are $(156, 90, 99)$ km s^{-1}, respectively. These values are broadly consistent with the mean velocities $(\tilde{v}_x, \tilde{v}_y, \tilde{v}_z) = (-32, -34, 13)$ km s^{-1}, and their associated velocity dispersion $(152, 130, 119)$ km s^{-1} of the halo sample in the TGAS × RAVE training set. The differences are likely due to the different number of stars, as well as the different sky coverage and radial range probed by the different samples. The halo sample selected from the TGAS × RAVE data extends

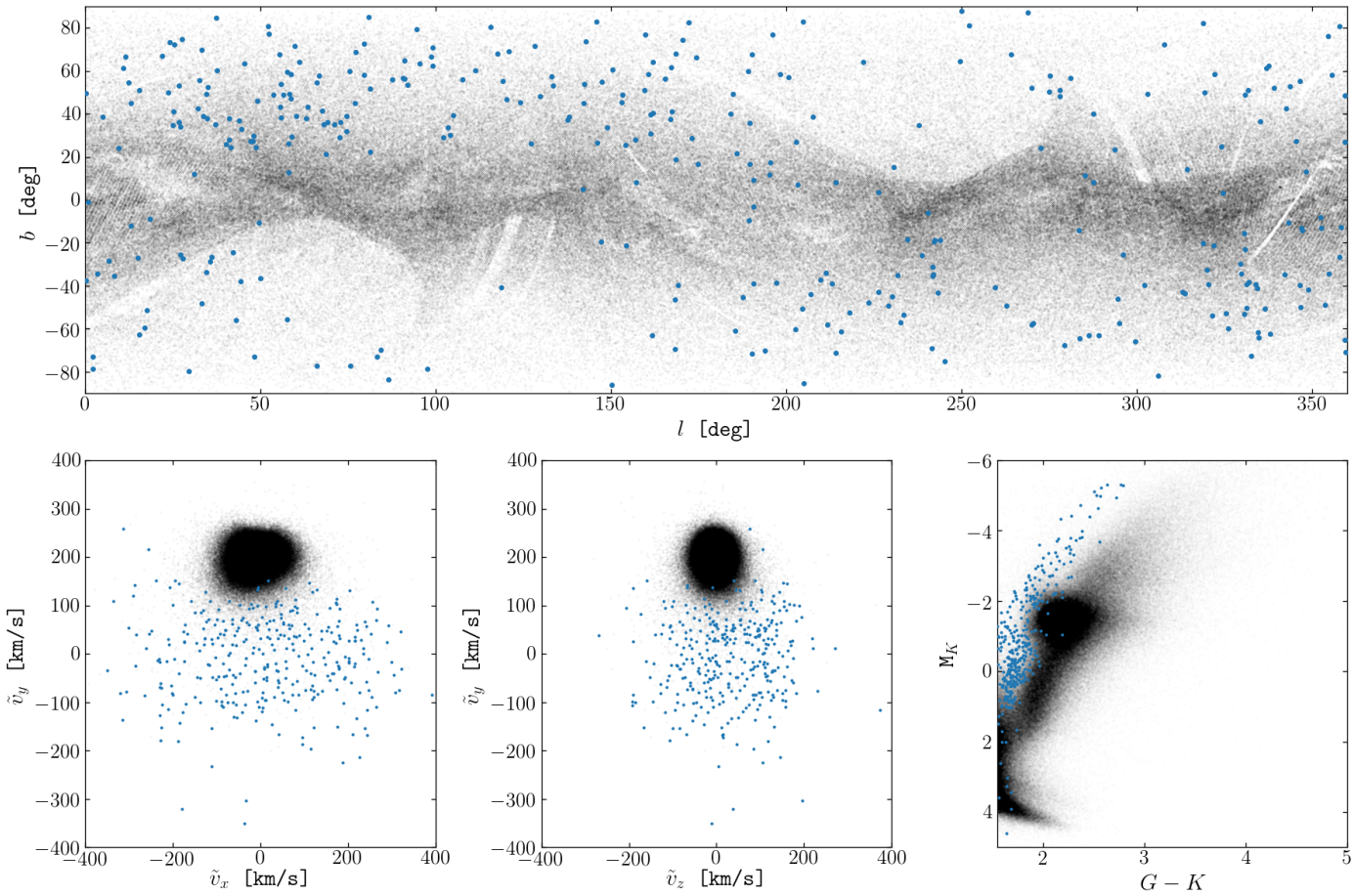

Fig. 8. On-sky positions, velocity distributions, and a HR diagrams of the TGAS data fed to the halo classifier. The stars have relative parallax uncertainties better than 30% and distances greater than 0.1 kpc. The 337 blue points mark the locations of the halo candidates, while the black dots are the rest of the stars. One can see that the halo candidates have kinematics consistent with them being halo, and are located in regions of the HR typically associated with metal-poor giants.

out to nearly 7 kpc and covers almost half of the sky, while the sample of halo stars identified from the TGAS data with our classifier extends only out to approx 1.3 kpc but covers the entire sky.

To further test the reliability of the classifier, we do the following sanity checks. Shown with red points in Fig. 9, there are 465 stars with $\tilde{v}_y < 50$ km s^{-1} that are not selected by our classifier despite having kinematics consistent with halo stars. We find that these stars have redder $G-K$ colours compared to the halo stars in the training set and stars selected to be halo by the classifier. It is possible that some of these may be in regions of high extinction, which makes their colours redder than expected. The other possibility is the existence of a halo population that is more metal-rich than our unsupervised, metallicity-based selection of halo stars in the training set (Bonaca et al. 2017).

We also notice that there is a considerable number of stars not classified as halo that have brighter M_K magnitudes than the dashed line on the top panel in Fig. 9. This region of the HR diagram is heavily populated by halo stars in the training set. If we look at their velocity distribution, we find them to have purely disk kinematics. Large parallax uncertainties are one likely cause for them moving to the bright region of the HR diagram. When obtaining the distance to a star by inverting a parallax with a large uncertainty, one may overestimate the distance since its probability distribution function has an extended tail towards large distances (see Appendix A in Helmi et al. 2017). Overestimating the distance leads to an overestimation of the absolute magnitude of a star, which may explain the location of these disk stars on the HR diagrams in Figs. 8 and 9.

It might also be useful to compare the results of our model to the recent study by Posti et al. (2018), who showed that one can identify halo stars with a physically motivated model, in which each Galactic component is described by a distribution function. One can also apply such a model in cases in which the line-of-sight velocity of the stars is not known by marginalizing over the missing observable.

When applying the dynamical model of Posti et al. (2018) to the TGAS dataset, we identify 1667 halo candidate stars. This larger number of candidates in comparison to our technique is likely driven by the dynamical model not using metallicity information and not being biased against the detection of halo dwarfs as in the case of our classifier, although it is affected by assumptions on the Milky Way's gravitational potential and the spatial and velocity distribution of each Galactic component. In an attempt to make a "fairer" comparison between the two procedures of identifying halo stars, we count the number of candidates identified by the dynamical model that satisfy $1.55 \leq (G-K) < 2$ and $M_K < 2$. These cuts essentially select the metal-poor giant stars, since our classifier has been trained to detect metal-poor halo stars and is practically insensitive to the existence of halo dwarfs due to the limitations of the training set. We find that 265 halo star candidates have been identified by

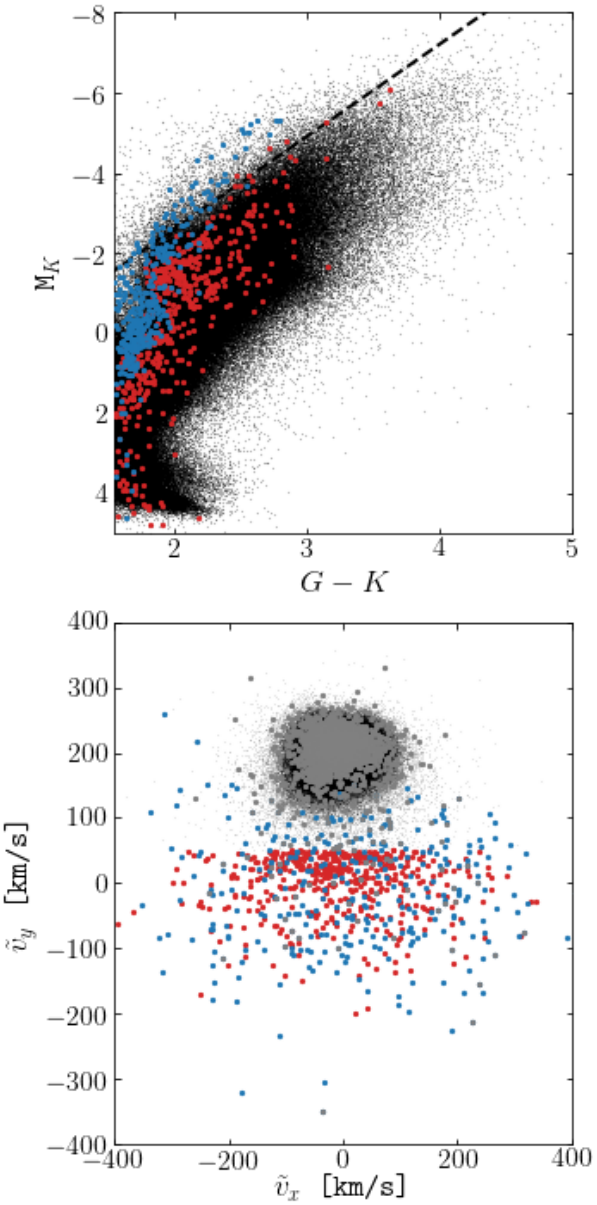

Fig. 9. Red points are TGAS stars that have $\tilde{v}_y < 50$ km s^{-1}and are not tagged as halo by our classifier, despite having halo kinematics. Those stars are found to have redder $G-K$ colours than expected for metal-poor halo stars. It is possible that these stars belong to the metal-poor tail of the thick disk. The grey points (*bottom panel*) have M_K magnitudes brighter than the cut marked by the black dashed line (*top panel*) and are not labelled as halo by our classifier. Even though they have colours consistent with those of metal-poor stars, the kinematics of the vast majority of these sources is purely consistent with that of disk stars.

the dynamical model that satisfy these conditions, a number very similar to the 277 candidates identified by our gradient boosted tree classifier given the same cuts.

5. Conclusions

The stellar halo is an essential component needed to understand the assembly process and some of the key properties of our Galaxy. A sizeable sample of halo stars can help us to constrain the Milky Way's mass or the shape of its gravitational potential. By searching for phase-space substructures in the stellar halo we may be able to isolate some of the primordial building blocks of the Galaxy.

In this contribution we trained a series of gradient boosted trees machine learning models and assessed their ability to identify halo stars in the publicly accessible *Gaia* DR1 and the upcoming *Gaia* DR2 catalogues, based on the available astrometric and photometric data. We first tested the performance of our models on a data sample selected from the *Gaia* Universe Model Snapshot (GUMS) with a TGAS-like selection function. When using the full phase-space information and the optical *Gaia* photometry, the model identifies over 90% of the halo stars in an entirely unseen dataset. When the training data lacks $v_{\rm los}$ information, the model recovers over 85% of the halo stars in an unseen test set. In both of these ideal cases, the level of contamination in the labelled halo sample is negligible (<1%).

To investigate the performance of our halo star classifier in more realistic cases, we convolved a large GUMS sample with stars that have $6 < G < 19$ with uncertainties expected for *Gaia* DR2. For stars with relative parallax errors <30%, the model is able to identify ~60% of the halo stars in the above magnitude range when using *Gaia* photometry and 5D phase-space information only as training features. For magnitudes brighter than $G \approx 12.5$, full astrometric solutions should be available in *Gaia* DR2, and coupling that with the optical photometry from the satellite we expect that it will be possible to correctly identify at least ~90% of the halo stars that have reasonable distances. From the above experiments, we conclude that even though a lack of $v_{\rm los}$ does decrease the performance of our classifier, we are still able to reliably identify a large fraction of halo stars, with a negligible level of contamination.

It is worth noting that our requirement that stars have a relative parallax uncertainty <30% considerably reduces the reach of our method. In our *Gaia* DR2-like sample selected from GUMS, 90% of the halo stars that satisfy this criterion are within 5 kpc from the Sun, and this constitutes only ~2% of the full halo population. Efforts that aim to improve parallax uncertainties (e.g. Anderson et al. 2018) may therefore be crucial and could lead to a much higher fraction of the halo being accessible. Engineering informative training features that do not require a direct use of the distance but could employ the parallax directly without introducing significant biases could in principle extend this method to samples spanning larger volumes.

In order to apply our halo classifier on the TGAS data, we trained it on a sample of halo stars identified in the TGAS × RAVE dataset published by McMillan et al. (2018), which contains updated spectrophotometric parallaxes that use the TGAS parallaxes as priors. The halo sample in the TGAS × RAVE was selected via a metallicity cut and it was further cleaned by unsupervised kinematic modelling, in an entirely data driven way. Here we focused on the stars that belong to the red giant branch since very few halo stars in the TGAS × RAVE data belong to the main sequence, and they are insufficient to properly train the model in that regime.

We applied the classifier trained on the TGAS × RAVE data on the entire TGAS catalogue, and identified 337 high confidence red giant branch halo stars. While this number may seem very small at first glance, it is because we only consider TGAS stars that have relative parallax errors smaller than 30%, which severely limits the number of red giant branch stars in the sample. Nevertheless the halo sample selected in this manner has broadly consistent kinematics with the halo sample selected from the TGAS × RAVE dataset, adding confidence that our model is performing correctly.

We look forward to applying our halo star classifier on the data that will come from the second data release of the *Gaia* mission. Our tests show that with this method we should be able to identify a high confidence halo sample that numbers in the thousands. Such a sample would be extremely valuable in helping us unravel the history of our Galaxy.

Acknowledgements. We would like to thank the anonymous referee for their comments which improved the manuscript. We thank Anthony Brown for his useful comments regarding the expected uncertainties in *Gaia* DR2. We are also very grateful to Yonatan Alexander for his advice regarding the usage for `XGBoost` and `bayes_opt`. This work has been supported by a VICI grant from the Netherlands Organisation for Scientific Research, NWO, and by NOVA, the Netherlands Research School for Astronomy. We have made use of data from the European Space Agency (ESA) mission *Gaia* (`http://www.cosmos.esa.int/gaia`), processed by the *Gaia* Data Processing and Analysis Consortium (DPAC, `http://www.cosmos.esa.int/web/gaia/dpac/consortium`). Funding for the DPAC has been provided by national institutions, in particular the institutions participating in the *Gaia* Multilateral Agreement.

In addition to the software discussed in the paper, this work made use of `vaex` (Breddels 2017), `numpy` (Walt et al. 2011), and `matplotlib` (Hunter 2007).

References

Anderson, L., Hogg, D. W., Leistedt, B., Price-Whelan, A. M., & Bovy, J. 2018, AJ, 156, 145
Balbinot, E., Yanny, B., Li, T. S., et al. 2016, ApJ, 820, 58
Bekkerman, R., Bilenko, M., & Langford, J. 2011, in Proc. of the 17th ACM SIGKDD International Conf. Tutorials (New York, NY, USA: ACM), 4:1
Bell, E. F., Zucker, D. B., Belokurov, V., et al. 2008, ApJ, 680, 295
Bell, E. F., Xue, X. X., Rix, H.-W., Ruhland, C., & Hogg, D. W. 2010, AJ, 140, 1850
Belokurov, V., Zucker, D. B., Evans, N. W., et al. 2006, ApJ, 642, L137
Bernard, E. J., Ferguson, A. M. N., Schlafly, E. F., et al. 2014, MNRAS, 443, L84
Bernard, E. J., Ferguson, A. M. N., Schlafly, E. F., et al. 2016, MNRAS, 463, 1759
Bonaca, A., Conroy, C., Wetzel, A., Hopkins, P. F., & Kereš D. 2017, ApJ, 845, 101
Bond, H. E. 1980, ApJS, 44, 517
Bovy, J., Bahmanyar, A., Fritz, T. K., & Kallivayalil, N. 2016, ApJ, 833, 31
Breddels, M. A. 2017, in Astroinformatics, eds. M. Brescia, S. G. Djorgovski, E. D. Feigelson, G. Longo, & S. Cavuoti, IAU Symp., 325, 299
Brochu, E., Cora, V. M., & de Freitas, N. 2010, ArXiv e-prints [arXiv:1012.2599]
Carney, B. W., Laird, J. B., Latham, D. W., & Aguilar, L. A. 1996, AJ, 112, 668
Chen, T., & Guestrin, C. 2016, ArXiv e-prints [arXiv:1603.02754]
Chiappini, C., Matteucci, F., & Romano, D. 2001, ApJ, 554, 1044
Cooper, A. P., Cole, S., Frenk, C. S., et al. 2010, MNRAS, 406, 744
Cover, T., & Hart, P. 2006, Inf. Theor., 13, 21
Crnojević, D., Sand, D. J., Spekkens, K., et al. 2016, ApJ, 823, 19
Deason, A. J., Belokurov, V., Evans, N. W., et al. 2012, MNRAS, 425, 2840
Deason, A. J., Belokurov, V., Koposov, S. E., et al. 2017, MNRAS, 470, 1259
Drake, A. J., Catelan, M., Djorgovski, S. G., et al. 2013a, ApJ, 763, 32
Drake, A. J., Catelan, M., Djorgovski, S. G., et al. 2013b, ApJ, 765, 154
Friedman, J. H. 2001, Ann. Stat., 29, 1189
Gaia Collaboration (Brown, A. G. A., et al.) 2016, A&A, 595, A2
Grillmair, C. J. 2006, ApJ, 645, L37
Grillmair, C. J., & Dionatos, O. 2006, ApJ, 643, L17
Helmi, A., & de Zeeuw P. T. 2000, MNRAS, 319, 657
Helmi, A., & White, S. D. M. 1999, MNRAS, 307, 495
Helmi, A., Cooper, A. P., White, S. D. M., et al. 2011, ApJ, 733, L7
Helmi, A., Veljanoski, J., Breddels, M. A., Tian, H., & Sales, L. V. 2017, A&A, 598, A58
Hunter, J. D. 2007, Comput. Sci. Eng., 9, 90
Ibata, R. A., Gilmore, G., & Irwin, M. J. 1994, Nature, 370, 194
Ibata, R. A., Gilmore, G., & Irwin, M. J. 1995, MNRAS, 277, 781
Ibata, R. A., Lewis, G. F., McConnachie, A. W., et al. 2014, ApJ, 780, 128
Kafle, P. R., Sharma, S., Robotham, A. S. G., et al. 2017, MNRAS, 470, 2959
Koposov, S. E., Belokurov, V., Evans, N. W., et al. 2012, ApJ, 750, 80
Kunder, A., Kordopatis, G., Steinmetz, M., et al. 2017, AJ, 153, 75
Law, D. R., & Majewski, S. R. 2010, ApJ, 718, 1128
Law, D. R., Majewski, S. R., & Johnston, K. V. 2009, ApJ, 703, L67
Li, P., Burges, C. J. C., & Wu, Q. 2007, NIPS'07 USA, Curran Associates Inc., 897
Lindegren, L., Lammers, U., Bastian, U., et al. 2016, A&A, 595, A4
Majewski, S. R. 1992, ApJS, 78, 87
Majewski, S. R., Munn, J. A., & Hawley, S. L. 1996, ApJ, 459, L73
Martin, N. F., Ibata, R. A., Rich, R. M., et al. 2014, ApJ, 787, 19
Martínez-Delgado, D., Peñarrubia, J., Gabany, R. J., et al. 2008, ApJ, 689, 184
Martínez-Delgado, D., Gabany, R. J., Crawford, K., et al. 2010, AJ, 140, 962
McConnachie, A. W., Irwin, M. J., Ibata, R. A., et al. 2009, Nature, 461, 66
McMillan, P. J., Kordopatis, G., Kunder, A., et al. 2018, MNRAS, 477, 5279
Morrison, H. L., Flynn, C., & Freeman, K. C. 1990, AJ, 100, 1191
Morrison, H. L., Mateo, M., Olszewski, E. W., et al. 2000, AJ, 119, 2254
Newberg, H. J., Yanny, B., Rockosi, C., et al. 2002, ApJ, 569, 245
Posti, L., Helmi, A., Veljanoski, J., & Breddels, M. 2018, A&A, 615, A70
Robin, A. C., Reylé, C., Derrière, S., & Picaud, S. 2003, A&A, 409, 523
Robin, A. C., Luri, X., Reylé, C., et al. 2012, A&A, 543, A100
Sanders, J. L. & Binney, J. 2013, MNRAS, 433, 1826
Searle, L., & Zinn, R. 1978, ApJ, 225, 357
Sesar, B., Ivezić, Ž., Grammer, S. H., et al. 2010, ApJ, 708, 717
Skrutskie, M. F., Cutri, R. M., Stiening, R., et al. 2006, AJ, 131, 1163
Slater, C. T., Bell, E. F., Schlafly, E. F., et al. 2013, ApJ, 762, 6
Smith, M. C., Evans, N. W., Belokurov, V., et al. 2009, MNRAS, 399, 1223
Snoek, J., Larochelle, H., & Adams, R. P. 2012, ArXiv e-prints [arXiv:1206.2944]
Starkenburg, E., Helmi, A., Morrison, H. L., et al. 2009, ApJ, 698, 567
Torrealba, G., Catelan, M., Drake, A. J., et al. 2015, MNRAS, 446, 2251
Tyree, S., Weinberger, K. Q., Agrawal, K., & Paykin, J. 2011, in Proc. of the 20th International Conference on World Wide Web (New York, NY, USA: ACM) 387
Walt, S. v. d., Colbert, S. C., & Varoquaux, G. 2011, Comput. Sci. Eng., 13, 22
Watkins, L. L., Evans, N. W., Belokurov, V., et al. 2009, MNRAS, 398, 1757
Xue, X. X., Rix, H. W., Zhao, G., et al. 2008, ApJ, 684, 1143
Xue, X.-X., Rix, H.-W., Yanny, B., et al. 2011, ApJ, 738, 79
Xue, X.-X., Ma, Z., Rix, H.-W., et al. 2014, ApJ, 784, 170