Monthly Notices
of the
ROYAL ASTRONOMICAL SOCIETY

Mon. Not. R. Astron. Soc. **427,** 1284–1297 (2012) doi:10.1111/j.1365-2966.2012.22061.x

An improved quasar detection method in EROS-2 and MACHO LMC data sets

K. Pichara,[1,2]⋆ P. Protopapas,[2,3] D.-W. Kim,[2,3] J.-B. Marquette[4] and P. Tisserand[5]

[1]*Computer Science Department, Pontificia Universidad Católica de Chile, Santiago 782-0436, Chile*
[2]*Institute for Applied Computational Science, Harvard University, Cambridge, MA 02138, USA*
[3]*Harvard–Smithsonian Center for Astrophysics, Cambridge, MA 02138, USA*
[4]*UPMC-CNRS, UMR7095, Institut d'Astrophysique de Paris, F-75014 Paris, France*
[5]*Research School of Astronomy and Astrophysics, Australian National University, Canberra, ACT 0200, Australia*

Accepted 2012 September 5. Received 2012 September 3; in original form 2012 June 15

ABSTRACT

We present a new classification method for quasar identification in the EROS-2 and MACHO data sets based on a boosted version of a random forest classifier. We use a set of variability features including parameters of a continuous autoregressive model. We prove that continuous autoregressive parameters are very important discriminators in the classification process. We create two training sets (one for EROS-2 and one for MACHO data sets) using known quasars found in the Large Magellanic Cloud (LMC). Our model's accuracy in both EROS-2 and MACHO training sets is about 90 per cent precision and 86 per cent recall, improving the state-of-the-art models, accuracy in quasar detection. We apply the model on the complete, including 28 million objects, EROS-2 and MACHO LMC data sets, finding 1160 and 2551 candidates, respectively. To further validate our list of candidates, we cross-matched our list with 663 previously known strong candidates, getting 74 per cent of matches for MACHO and 40 per cent in EROS.

The main difference on matching level is because EROS-2 is a slightly shallower survey which translates to significantly lower signal-to-noise ratio light curves.

Key words: methods: data analysis – Magellanic Clouds – quasars: general.

1 INTRODUCTION

Given the immense amount of data being produced by current deep-sky surveys such as Pan-STARRS (Kaiser et al. 2002), and Ivezic 2008 future surveys such as LSST (Ivezic 2008) and SkyMapper (Keller et al. 2007), astronomy is facing new challenges in how to analyse *big data* and thus in how to search or predict events/patterns of interest.

The size of the data has already exceeded the capability of manual examination or the capability of standard data analysis tools. LSST will produce 15 terabytes of data per night, which is even beyond the capacity of typical data storage today.

Thus, in order to analyse such a huge amounts of data and detect interesting events or patterns with minimum false positives, innovative and novel data analysis methods are crucial for the success of such surveys.

In our previous works (Kim et al. 2011a, 2012), we developed classification models for the selection of quasars from large photometric data bases using variability characteristics as the main discriminators. In particular, we used a supervised classification model trained using a set of variability features calculated from MACHO light curves (Alcock et al. 2000). We applied the trained model to the entire MACHO data base of ∼40 million light curves and selected a few thousand quasar candidates. In this paper, we present an improved classification model used to detect quasars on MACHO (Alcock et al. 2000) and EROS-2 data set (Tisserand et al. 2007). The new model, which works over an extended set of variability features, substantially decreases false positive rate and increases efficiency.

The actual model improvement is a result of an improvement in the machine learning classification model and the light-curve features we use. Machine learning classification methods have been very popular for many decades. These methods are data analysis models that learn to predict a categorical variable from a set of other variables (of any type). Most known classification models are decision trees (Quinlan 1993), naive Bayes (Duda & Hart 1973), neural networks (Rumelhart, Hinton & Williams 1986), support vector machines (SVMs; Cortes & Vapnik 1995) and random forest (RF; Breiman 2001). There are some meta-models to improve classification results such as boosting methods (Freund & Schapire 1997) and mixtures of experts (Jordan 1994), among others. In general, more recent classifiers are a result of research focused on building models able to search for patterns within high-dimensional data sets, where the combinatorial number of possible projections of data is large.

⋆E-mail: kpb@ing.puc.cl

Many machine learning classifiers have been applied to the analysis of astronomical data, in particular to classify transients and variable stars from time series data (Debosscher et al. 2007; Wachman et al. 2009; Wang, Khardon & Protopapas 2010; Bloom & Richards 2011; Bloom et al. 2011; Kim et al. 2011a,b; Richards et al. 2011). Wang et al. (2010) proposed an algorithm to fit phase-shifted periodic time series using a mixture of Gaussian processes. Debosscher et al. (2007) used many machine learning classifiers to learn a model that classifies variable stars in a sample from *Hipparcos* and OGLE data bases. Richards et al. (2011) used an RF classifier to classify between pulsational variables and eclipsing systems used in Milky Way tomography. In Bloom et al. (2011), machine learning algorithms are used to classify transients and variable stars from the Palomar Transient Factory survey (Rau et al. 2009). Wachman et al. (2009) used cross-correlation as a phase-invariant feature to be used as a similarity indicator in a kernel function.

In this work, we used an RF classifier (Breiman 2001) boosted with the AdaBoost algorithm (Freund & Schapire 1997). The RF classifier comes from the well-known decision tree model (Quinlan 1993) and bagging techniques (Breiman 1996), where the model randomly explores several subsets of features while analysing samples of training data. This model performs very well in many machine learning domains (Breiman 2001). The AdaBoost algorithm (Freund & Schapire 1997) is a boosting technique which fits a sequence of classification models (in this case a sequence of many RFs) to different subsets of data objects (in our case light curves), generating a mixture of classifiers, each one specialized in smaller areas of the feature space. We call these classifiers 'weak classifiers' or 'simpler classifiers'. This is a nice property for quasar classification, given that there are only a few known training quasars compared with the amount of non-quasars light curves. Having some weak classifiers that take care of some areas with no training quasars helps us to filter out many non-quasars, while other specialized classifiers perform well near the quasar areas in the feature space.

Besides improving the classification model, we added new features as descriptors of light curves. These features correspond to the parameters of the continuous autoregressive [CAR(1)] model (Belcher, Hampton & Wilson 1994) fitted to the light curves. Previous work shows that describing quasars using CAR(1) fitting parameters gives suitable results to differentiate them from other classes of light curves (Kelly, Bechtold & Siemiginowska 2009). Kelly et al. (2009) did not use machine learning classifiers to automatically detect quasars; they use a CAR(1) model to fit 100 quasar light curves in order to find correlations between CAR(1) parameters and luminosity characteristics.

In our work, we show that by adding CAR(1) features to our previous set of features (used in Kim et al. 2011a), we can learn more accurate models for quasar detection. Given that our model is built to find quasars over dozens of millions of stars, we need to be very efficient in the estimation of the optimal parameters in order to make the process feasible within a considerable amount of time. Unfortunately, methods such as Metropolis–Hastings or Gibbs sampling are not suitable for our purposes, given the computational cost they involve.

To gain efficiency, we reduce the problem by approximating one of the parameters (the mean value of the light curve) and optimizing the remaining parameters (the amplitude and time-scale of the variability) using a multidimensional unconstrained nonlinear minimization (Nelder & Mead 1965). Once we get the optimal parameters, we use them as features of the object corresponding to the light curve. Besides the CAR(1) features, we also used time series features as in our previous work (Kim et al. 2011b); in Section 4 we give details about all the features we extracted.

To check the fitting accuracy of our model, we first calculate the training accuracy of our classifier using 10-fold cross-validation over a training set, which consists of about 6000 known light curves corresponding to different kinds of variable stars, non-variable stars and confirmed quasars, one set corresponding to the MACHO data base and another to the EROS-2 data base. In the MACHO case, we substantially improve our training accuracy compared with our previous work (Kim et al. 2011b), increasing 14.3 per cent in precision and 3.6 per cent in recall for the MACHO data base. In the EROS-2 training data base, we get about the same training efficiency as in the MACHO case but we could not compare it to our previous work because this is the first time we attempt to classify in the EROS-2 data base. As an extra test for our candidates, we cross-match them with the previous set of strong candidates found in Kim et al. (2011b); details are presented in Section 5.

Using parallel computing, we decrease the processing time to allow us to select quasar candidates from the entire data base within three days. Note that the data analysis schema used in this work can be applied to any of the ongoing and future synoptic sky surveys such as Pan-STARRS, LSST and SkyMapper, among others.[1]

If confirmed, the selected quasars from the MACHO data base will provide critical information for galaxy evolution, black hole growth, large-scale structure, etc. (Heckman et al. 2004; Bower et al. 2006; Trichas et al. 2009, 2010). Moreover, the resulting quasar light curves will be a valuable data set for quasar time variability studies (e.g. time-scale, black hole mass, type I and II variability) since MACHO (Alcock et al. 2000) and EROS light curves are well sampled over 7.4 years after the poineering search for QSO with the EROS-1 data set (Beaulieu et al. 1996).

The paper is organized as follows. In Section 2, we present details about the EROS-2 data base. In Section 3, we describe in detail the classification model we use, including the RF model and AdaBoost. In Section 4, we describe the features we use to describe the light curves, and in Section 5 we describe the experimental results for the MACHO and EROS-2 data sets.

2 EROS-2 DATA SET

The EROS-2 collaboration made use of the MarLy telescope, a 1-m diameter Ritchey–Chrétien (f/5.14) instrument dedicated to the survey. It was operated between 1996 July and 2003 March at La Silla Observatory (ESO, Chile). It was equipped with two wide-angle CCD cameras which are located behind a dichroic beam splitter. Each camera is a mosaic of eight CCDs, two along right ascension and four along declination. Each CCD has 2048 × 2048 pixel of 15 × 15 μm^2 individual size, corresponding to a 0.6 × 0.6 $arcsec^2$ pixel surface on the sky. The size of the field of view is $0\overset{\circ}{.}7$ along right ascension and $1\overset{\circ}{.}4$ along declination. The dichroic beam splitter allowed simultaneous imaging in two broad non-standard passbands, B_E in the range 4200–7200 (the so-called 'blue' channel) and R_E in the range 6200–9200 (the so-called 'red' channel). The blue filter is intermediate between the standard V and R standard passbands, while the red filter is analogous to I_c. The normalized transmission curve of these filters, compared to standard ones, is given by Hamadache (2004, their fig. 3.3)[2]. Tisserand

[1] Our main computer resource is the Odyssey cluster supported by the FAS Research Computing Group at Harvard.

[2] Available at http://tel.archives-ouvertes.fr.

et al. (2007, equation 4) give the equations to transform EROS-2 magnitudes into V and I_c ones within an accuracy of 0.1 mag.

The light curves of individual stars were constructed from fixed positions on templates using PEIDA, a software specifically developed for the photometry of EROS-2 images (Ansari 1996). The nomenclature of objects is as defined in Derue et al. (2002).

3 METHODOLOGY

To train a model that learns to detect quasars, we propose to use a combination of classifiers. Combination of multiple classifiers was first proposed by Xu, Krzyzak & Suen (1992). In that work, they proved that combining multiple classifiers overcomes many of the individual classifier limitations. In many pattern recognition problems, such as character recognition, handwritten text recognition and face recognition (Plamondon & Srihari 2000; Zhao et al. 2003), a combination of multiple classifiers obtains much better classification performance. One effective way to combine classifiers is the AdaBoost algorithm, proposed in Freund & Schapire (1997).

The AdaBoost algorithm consists of a set of base classifiers that are trained sequentially, such that each classifier is trained on the instances where the previous classifier obtained a bad performance (learn what your partners could not learn). Freund & Schapire (1997) show that if the training set used for each classifier depends on the goodness of fit of the previous classifier, then the performance of the whole system improves. To make the base classifiers focus on different subsets of the training set, we assign weights to training data instances. The lower the weight for an instance, the less the classifier focuses on it (see Section 3.1 for further details).

One of the advantages of boosting methods is that after the model fitting phase is completed, each of the base classifiers becomes an expert in some subset of data objects. This is one of the main reasons that motivates us to use a previous boosting step. Given that we have a very small amount of known quasars in our training set compared with the amount of non-quasars, training a set of base classifiers that just learned how to filter out some of the non-quasars would be very helpful for the next base classifier used in the sequential process. We now present a detailed description of the boosting method we use in this work, the AdaBoost algorithm (Freund & Schapire 1997).

3.1 AdaBoost algorithm

AdaBoost, short for adaptive boosting, is a machine learning algorithm proposed by Freund & Schapire (1997). It is a meta-algorithm because it combines many learning algorithms to perform classification. AdaBoost is adaptive in the sense that subsequent classifiers built are tweaked in favour of those instances misclassified by previous classifiers. Although AdaBoost is sensitive to noisy data and outliers, it is less susceptible to overfitting (Dietterich 1995) than most learning algorithms.

In the context of light-curve classification, suppose we have a training (labelled) set of n light curves and q features describing each light curve. Each light curve in the training set has a known given label (e.g. quasar or non-quasar). Let $[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]$ be a set of n descriptors, where each $\boldsymbol{x}_i$, $i \in [1, \ldots, n]$, is a vector associated with the light curve i where its descriptor (features) values are $\{x_{i1}, \ldots, x_{iq}\}$, where q is the number of features. Let $\{y_1, \ldots, y_n\}$ be the labels such that $y_i = 1$ if the light curve i is a quasar and $y_i = -1$ otherwise.

Let H be the set of m classifiers $\{h_1, \ldots, h_m\}$, where $h_i: X \rightarrow Y$, and $D^{(t)}$ be the distribution of weights on classifiers at iteration t. Define m to be the number of classifiers and a constant T to be the number of times to iterate in the AdaBoost algorithm.

Initialization:

$X = [x_1, x_2, \ldots, x_n]$
$Y = [y_1, y_2, \ldots, y_n]$
$D^{(1)} = \left[d_1^{(1)}, d_2^{(1)}, \ldots, d_n^{(1)}\right] := \left[\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right]$
$T \leq n$

Algorithm:

for $t = 1$ to T **do**
  for $j = 1$ to m **do**
    $\epsilon_j := \sum_{i=1}^{n} d_i^{(t)} (1 - \delta_{y_i, h_j(x_i)})$
  end for
  $\epsilon_t := \min \epsilon_j$
  if $\epsilon_t \geq 0.5$ **then**
    break
  end if
  $h_t := \underset{h_j \in H}{\operatorname{argmin}} \{\epsilon_j\}$
  $\alpha_t := \frac{1}{2} \ln((1 - \epsilon_t)/\epsilon_t)$
  for $i = 1$ to n **do**
    $d_i^{(t+1)} := d_i^{(t)} \exp(-\alpha_t\, y_i\, h_t(x_i))/Z_t$
  end for
end for
$\mathcal{H}(X) := [\mathcal{H}(x_1), \mathcal{H}(x_2), \ldots, \mathcal{H}(x_n)]$, such that

$$\mathcal{H}(x_i) = \operatorname{sign}\left(\sum_{t=1}^{T} \alpha_t\, h_t(x_i)\right).$$

Notes:

(i) $\delta_{i,j}$ is the Kronecker delta;
(ii) Z_t is a normalization factor

$$Z_t = \sum_{i=1}^{n} d_i^{(t)} \exp(-\alpha_t\, y_i\, h_t(x_i)).$$

The equation to update the classifier weight distribution is constructed so that $-\alpha\, y_i\, h_t(x_i) < 1$ when $y_i = h_t(x_i)$ and $-\alpha\, y_i\, h_t(x_i) > 1$ when $y_i \neq h_t(x_i)$. Thus, after selecting an optimal classifier h_t, for the distribution D_t, the objects x_i that classifier h_t classified correctly are given less weight and those that it identified incorrectly are given more weight. Hence, when the algorithm proceeds to test the classifiers on $D^{(t+1)}$, it is more likely to select a classifier that better classifies the objects that h_t missed. AdaBoost minimizes the training error (exponentially fast) if each weak classifier performs better than random guessing ($\epsilon_t < 0.5$).

The base classifier we used in this work is the RF classifier (Breiman 2001), a very strong classifier that has shown very good results in many different domains. The following section shows details about the RF classifier.

3.2 Random Forest classifier

RF is a popular and very efficient algorithm based on decision tree models (Quinlan 1993) and bagging for classification problems (Breiman 1996, 2001). It belongs to the family of ensemble methods, appearing in machine learning literature at the end of the 1990s (Dietterich 2000), and has been used recently in the astronomical

journals (Carliles et al. 2010; Richards et al. 2011). The process of training or building an RF given training data is as follows.

(i) Let P be the number of trees in the forest and F the number of features on each tree; both values are model parameters.
(ii) Build P sets of n samples taken with replacement from the training set; this is called bagging. Note that each of the P bags has the same number of elements from the training set but fewer different examples, given that the samples are taken with replacement.
(iii) For each of the P sets, train a decision tree using a random sample of F features from the set of q possible features.

The RF classifier creates many linear separators inside many feature subsets until it gets suitable separations between objects from different classes. Linear separations come from each decision tree and each of the feature subsets comes from the random feature selection process on each tree. The bagging procedure is very useful to estimate the error of the classifier during the training process. This error can be estimated using out-of-the-bag procedure, which means 'evaluating the performance of each tree using the objects not selected in the bag which belong to the tree' (see Breiman 2001, for further details).

After training the RF, to classify a new unknown light-curve descriptor, one uses each of the decision trees already trained with the RF to classify the new unknown instance and the final decision is the most voted class among the set of P decision trees (see Breiman 2001, for more details). Breiman (2001) show that as the number of trees tends to infinity the classification error of the RF becomes bounded and the classifier does not overfit the data.

4 FEATURE EXTRACTION

We extracted 14 features per band for each light curve. These features correspond to 11 time series features used in our previous work (Kim et al. 2011b) and three features corresponding to the CAR(1) process.

4.1 Time series features

Here we very briefly summarize the 11 time series features used in our previous work (Kim et al. 2011b).

(i) $N_{\rm above}$, $N_{\rm below}$: the number of points above/below the upper/lower bound line calculated as points that are $\pm 4\sigma$ over the average of the autocorrelation functions.
(ii) Stetson $K_{\rm AC}$: the variability index derived based on the autocorrelation function of each light curve (Stetson 1996).
(iii) $R_{\rm cs}$: the range of the cumulative sums (starting from 1 to the number of observations) of each light curve (Ellaway 1978).
(iv) $\sigma/\bar{m}$: the ratio of the standard deviation, σ, to the mean magnitude, $\bar{m}$.
(v) Period and period S/N: using the Lomb–Scargle algorithm (Lomb 1976; Scargle 1982) we used the period with the highest value in the periodogram along with the signal-to-noise ratio of the best period.
(vi) Stetson L: a variability index (Stetson 1996) that describes the synchronous variability of different bands.
(vii) η: the ratio of the mean of the square of successive differences to the variance of data points.
(viii) $B - R$: average colour for each light curve.
(ix) Con: the number of three consecutive data points that are brighter or fainter than 2σ, normalized by $N - 2$.

4.2 Continuous autoregressive process features

We use continuous time autoregressive model [CAR(1)] to model irregular sampled time series in MACHO and EROS-2 light curves. CAR(1) process has three parameters, and it provides a natural and consistent way of estimating a characteristic time-scale and variance of light curves. CAR(1) process is described by the following

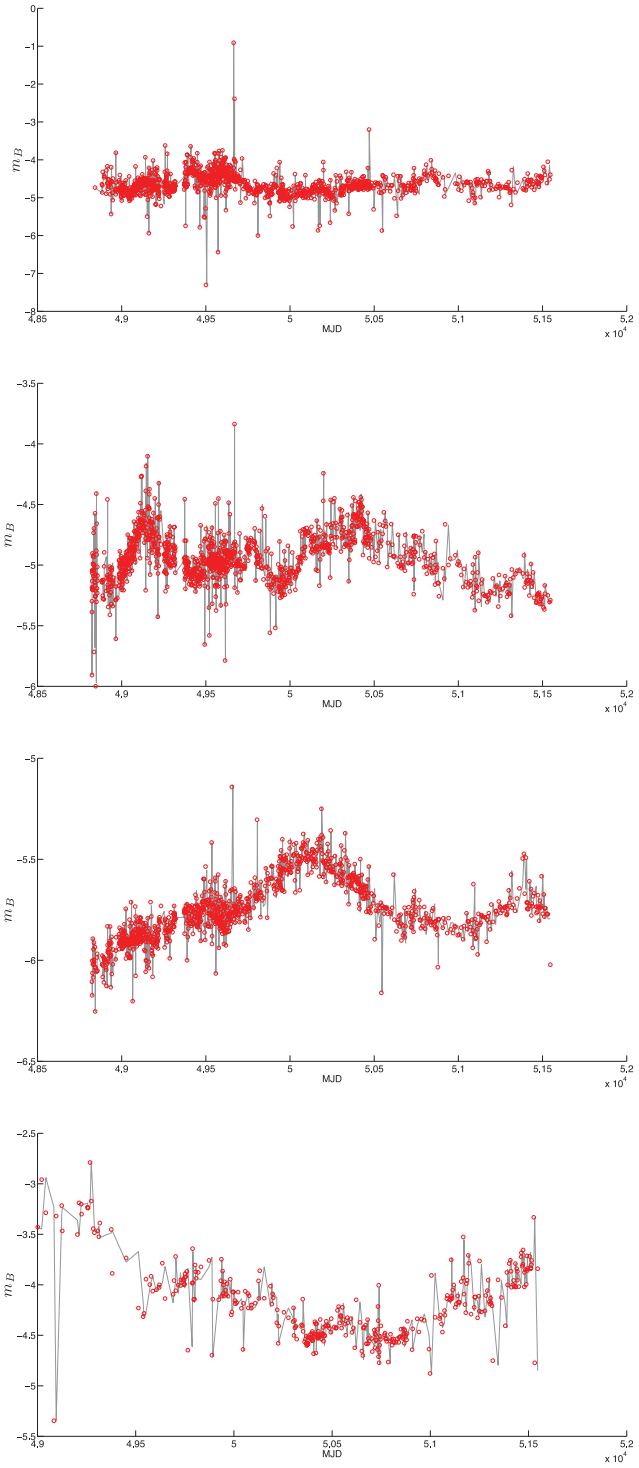

Figure 1. Quasar light curves (red circles) fitted with optimal CAR(1) model (grey lines) using the Nelder–Mead algorithm.

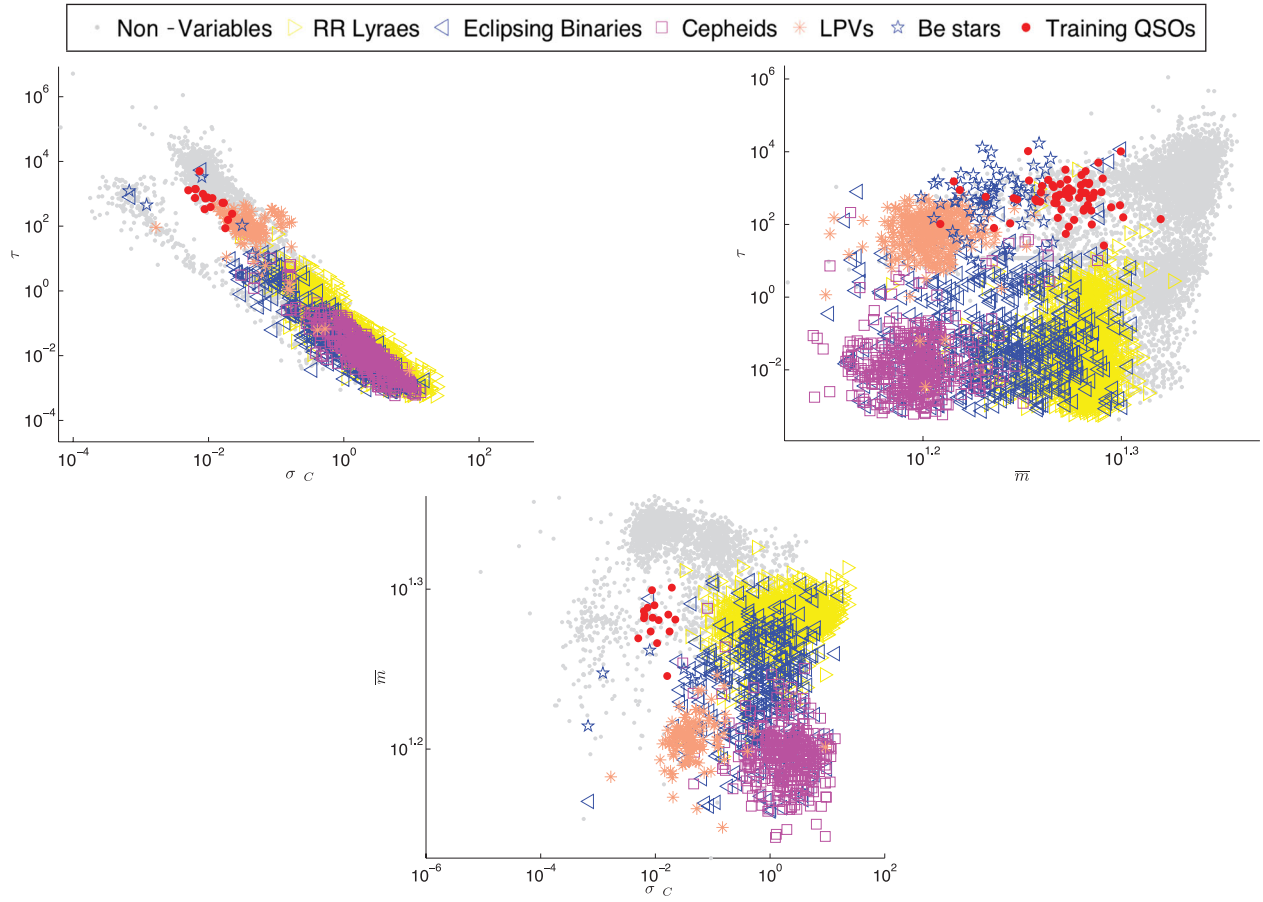

Figure 2. Projections on different pairs of CAR(1) features for EROS-2 training data.

stochastic differential equation (Brockwell & Davis 2002):

$$\mathrm{d}X(t) = -\frac{1}{\tau}X(t)\mathrm{d}t + \sigma_{\mathrm{C}}\sqrt{\mathrm{d}t}\,\epsilon(t) + b\,\mathrm{d}t, \quad \text{for} \quad \tau, \sigma_{\mathrm{C}}, t \geq 0\,, \tag{1}$$

where the mean value of the light curve $X(t)$ is $b\,\tau$ and the variance is $\tau\,\sigma_{\mathrm{C}}^2/2$. τ is the relaxation time of the process $X(t)$, and it can be interpreted as describing the variability amplitude of the time series. σ_{C} can be interpreted as describing the variability of the time series on time-scales shorter than τ. $\epsilon(t)$ is a white noise process with zero mean and variance equal to 1. The likelihood function of a CAR(1) model for a light curve with observations $\boldsymbol{x} = \{x_1, \ldots, x_n\}$ observed at times $\{t_1, \ldots, t_n\}$ with measurement error variances $\{\delta_1^2, \ldots, \delta_n^2\}$ is

$$p(\boldsymbol{x}|b, \sigma_{\mathrm{C}}, \tau) = \prod_{i=1}^{n} \frac{1}{[2\pi(\Omega_i + \delta_i^2)]^{1/2}} \exp\left\{-\frac{1}{2}\frac{(\hat{x}_i - x_i^*)^2}{\Omega_i + \delta_i^2}\right\}, \tag{2}$$

$$x_i^* = x_i - b\,\tau\,, \tag{3}$$

$$\hat{x}_0 = 0\,, \tag{4}$$

$$\Omega_0 = \frac{\tau\sigma_{\mathrm{C}}^2}{2}\,, \tag{5}$$

$$\hat{x}_i = a_i\hat{x}_{i-1} + \frac{a_i\Omega_{i-1}}{\Omega_{i-1} + \delta_{i-1}^2}(x_{i-1}^* + \hat{x}_{i-1})\,, \tag{6}$$

$$\Omega_i = \Omega_0\left(1 - a_i^2\right) + a_i^2\Omega_{i-1}\left(1 - \frac{\Omega_{i-1}}{\Omega_{i-1} + \delta_{i-1}^2}\right)\,, \tag{7}$$

$$a_i = \mathrm{e}^{-(t_i - t_{i-1})/\tau}\,. \tag{8}$$

To find the optimal parameters, we maximize the likelihood with respect to σ_{C}, b and τ. Given that the likelihood does not have an analytical solution, we can solve it with a statistical sampling method such as Metropolis–Hastings (Metropolis et al. 1953). Because we extract features for all the light curves in EROS-2 and MACHO data sets (about 28 and 40 millions of stars, respectively), performing a statistical sampling process to determine the optimal parameters would be feasible only in cases where stable solutions are found in a reasonable amount of time. We consider that less than 3 s is reasonable given our hardware resources. Unfortunately, we could not get stable solutions considering that restriction. To overcome this situation, we simplify the optimization problem by reducing the number of parameters to be estimated. Instead of estimating σ_{C}, b and τ, we just estimate σ_{C} and τ and then we calculate b as the mean magnitude of the light curve divided by τ. To check that this estimation works well, we use a sample of 250 light curves and compare the reduced χ^2 error using two- and three-parameter optimization, getting differences smaller than 2.5 per cent in average.

This approximation allows us to perform a two-dimensional optimization which can be solved with a regular numerical method in less than 1 s per light curve. We used the Nelder–Mead multidimensional unconstrained non-linear optimization (Nelder &

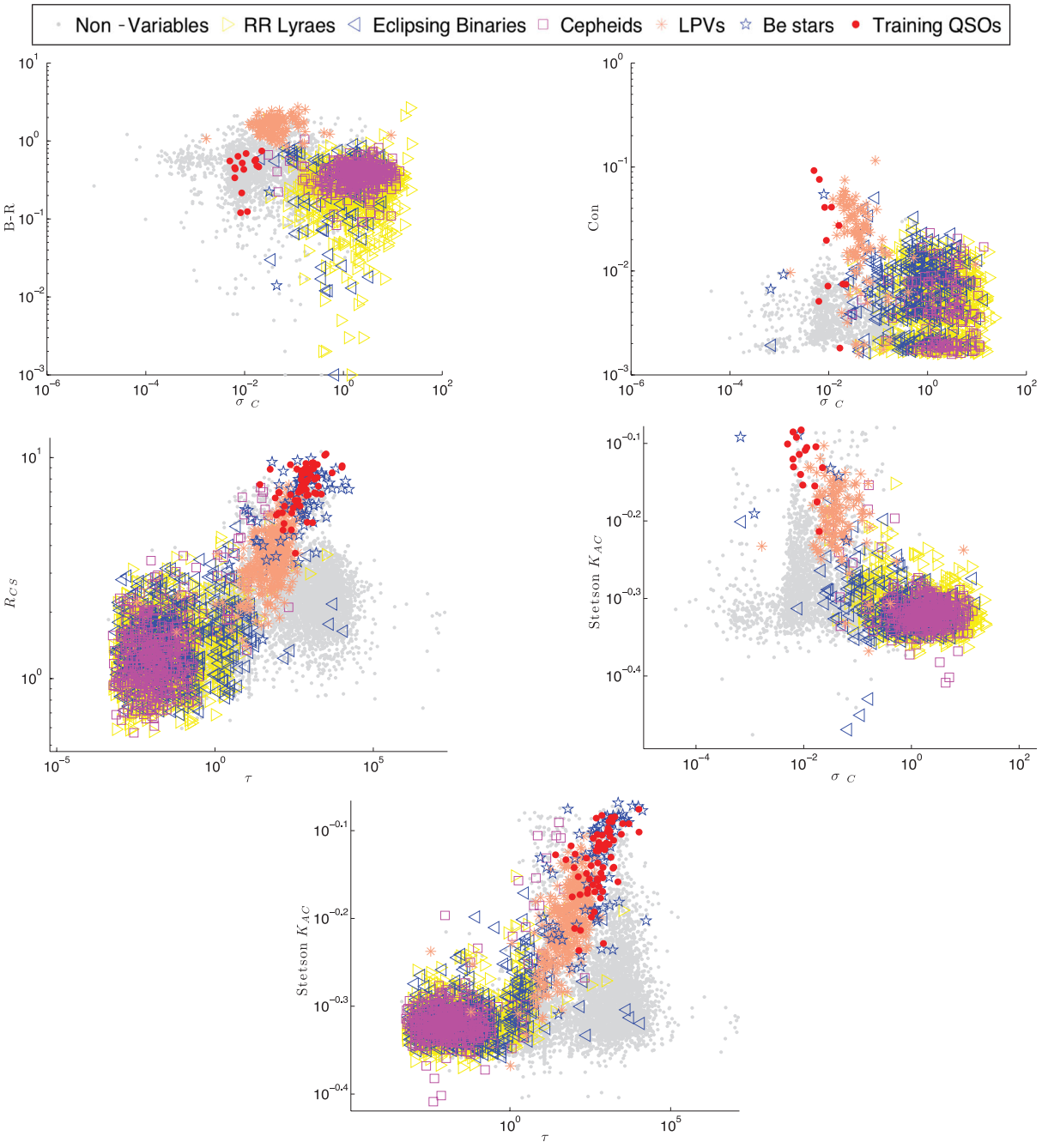

Figure 3. Projections on different pairs of features, combining CAR(1) features with time series features for EROS-2 training data.

Mead 1965) to find the optimal parameters. Fig. 1 shows the fitting of three quasar light curves with the resulting CAR(1) coefficients using the Nelder–Mead algorithm. Note that instead of using b directly as a feature, we use the mean magnitude of the light curve ($\overline{m}$), in order to have a cleaner feature (b is calculated from τ, which is already used as a feature).

5 QSO CANDIDATES ON EROS-2 AND MACHO DATA SETS

5.1 EROS-2 data set

To train a model capable of finding quasars in EROS-2, we create a training set composed of 65 known quasars, 67 Be stars, 330 long periodic stars, 5829 non-variable stars, 1727 RR Lyrae, 406 Cepheids and 488 EB stars. We get these stars by cross-matching the EROS-2 data set with MACHO known stars using positional matching with 3 arcsec of accuracy. We extracted features in bands R and B. Figs 2 and 3 show projections of the training set on different sets of features containing CAR(1) features. In many cases it is easy to get a natural separation between quasars and the variable stars, but usually quasars overlap many of the non-variable stars (e.g. σ_C with $B-R$, σ_C with τ, $\overline{m}$ with τ). Fortunately, there are many projections where quasars and non-variable stars are mostly separated (e.g. σ_C with Con, σ_C with $\overline{m}$, σ_C with Stetson K_{AC}, τ with R_{cs}, τ with Stetson K_{AC}).

To compare the distribution of the objects predicted as quasars with the training quasars and other variable stars, we plot our

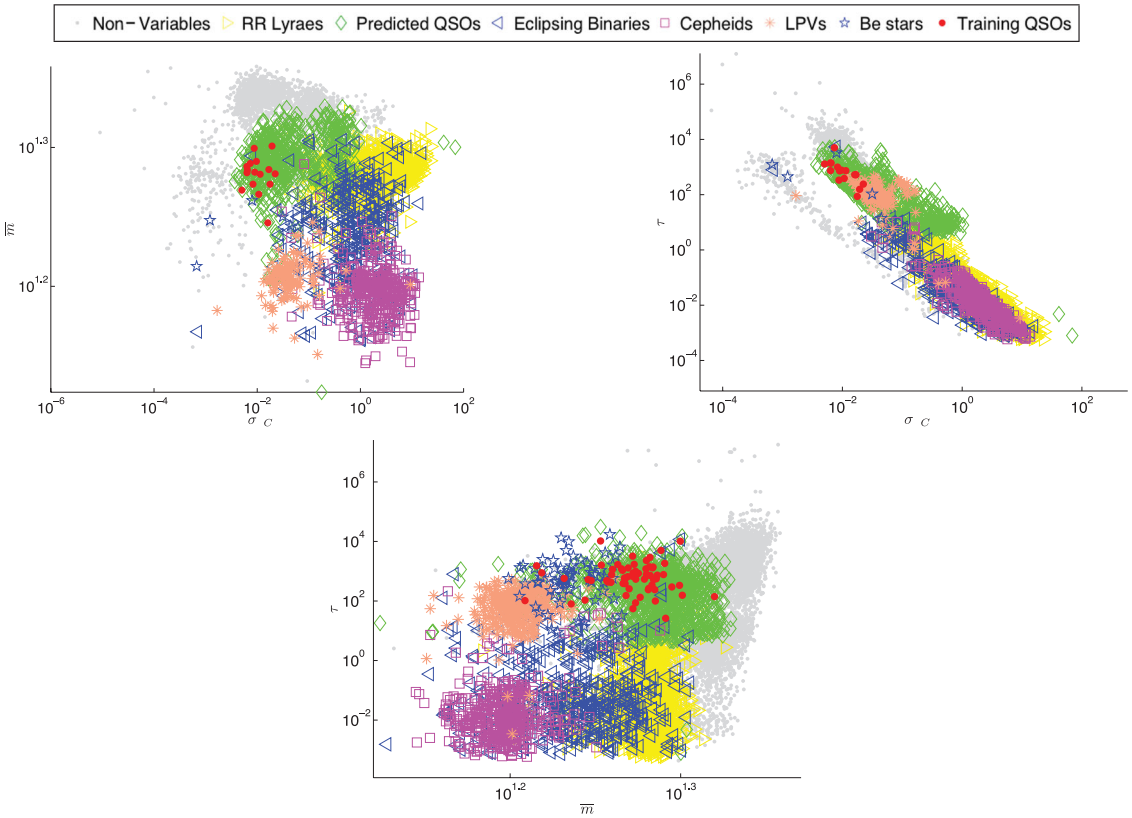

Figure 4. Predicted quasar and training star distributions projected on different pairs of CAR(1) features for EROS-2 data.

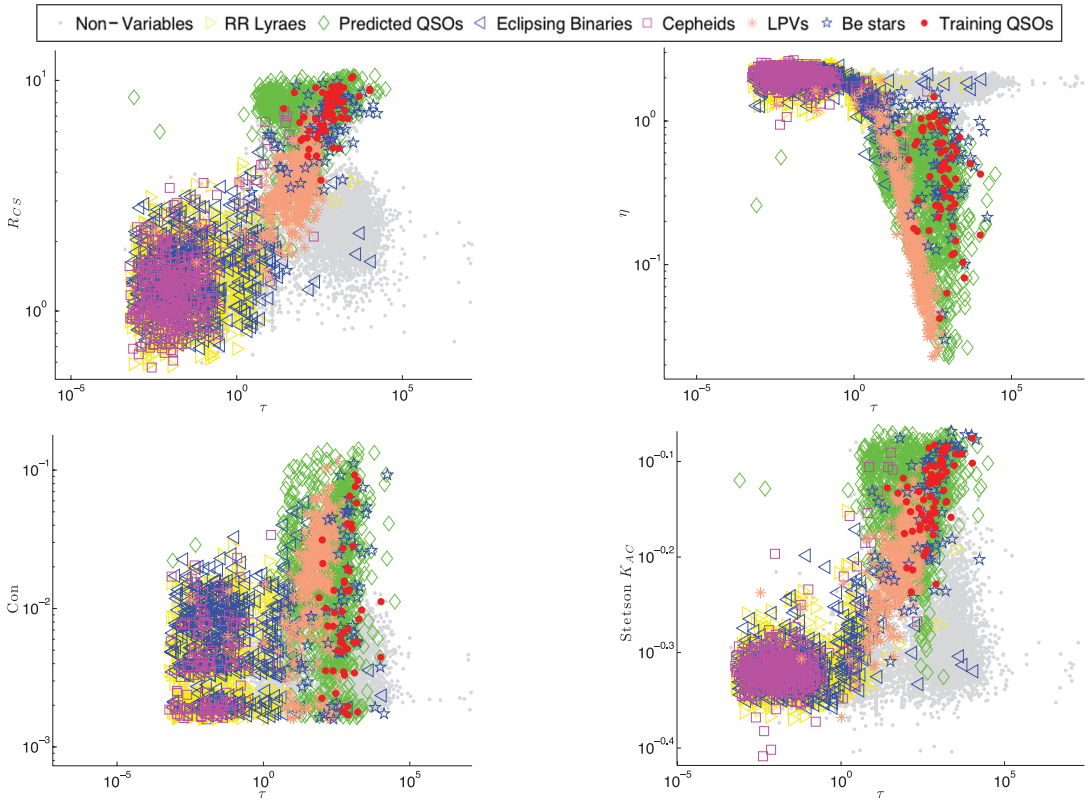

Figure 5. Predicted quasar and training star distributions for τ combined with three time series features for EROS-2 data.

Table 1. F-score for the EROS-2 training set using 10-fold cross-validation for different classification models. Each classifier is tuned with the optimal set of parameters. We can see that the boosted version of RF with CAR features outperforms other classification models. In all cases, using CAR features improves the result of the corresponding classifier.

SVM No CAR	SVM CAR	RF No CAR	RF CAR	AB+RF No CAR	AB+RF CAR
0.74	0.855	0.787	0.813	0.81	0.868

EROS-2 training data plus the predicted quasars projected on many different pairs of features (Figs 4 and 5). We can see that in most of the cases predicted quasars and training quasars have very similar distributions regardless of the small amount of training quasars we use. The main differences between both distributions are in general because of the big difference in size of the training and testing data, resulting in a set of predicted quasars ∼20 times bigger than the training quasars set.

To get an indicator of the accuracy in the training set on EROS-2 data set, we run a 10-fold cross-validation. This validation method consists of partitioning the data set in 10 folds (subsets) of the same size; we iterate 10 times and on iteration k we train the classifier with all the folds but the fold k, and then we test the performance on the fold k (the one which the model did not see during the training). The process returns the model prediction for the entire data set (the union of the 10 testing folds is equal to the original set). We measure the accuracy using the F-score indicator. This indicator is calculated as the harmonic mean of precision and recall:

$$F\text{-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}},$$

where precision and recall are defined as

$$\text{precision} = \frac{tp}{tp + fp} \qquad \text{recall} = \frac{tp}{tp + fn},$$

and tp, fp and fn are the number of true positives, false positives and false negatives, respectively.

Table 1 shows the results for the boosted version of RF, regular RF and SVM (classifier used in our previous work Kim et al. 2011b) with and without CAR features.

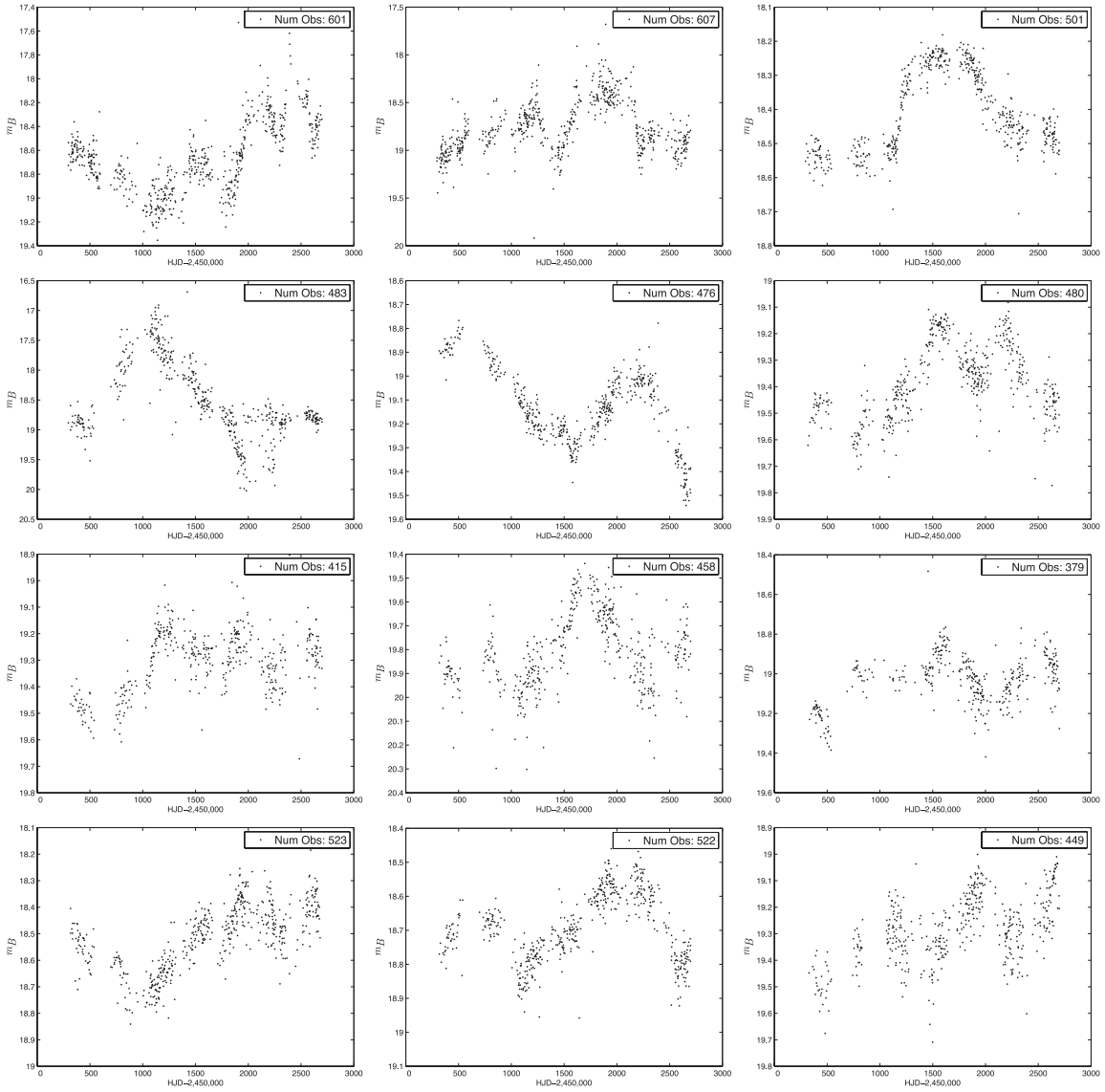

Figure 6. Light curves of quasar candidates predicted on EROS-2 data set.

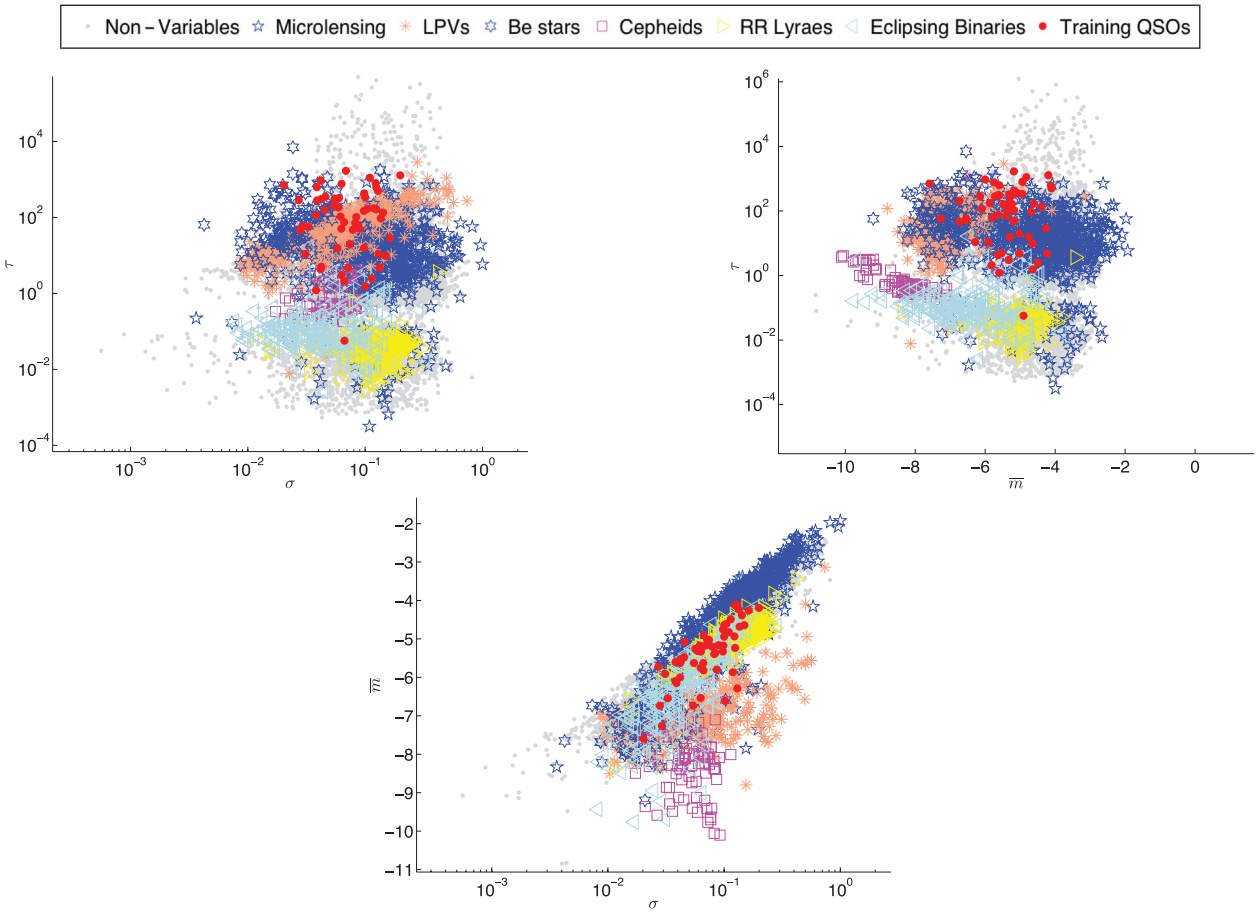

Figure 7. Training set projected on different pairs of CAR(1) features for MACHO data.

We find 1160 candidates in the EROS-2 data set. To validate our candidates, we cross-match them with the list of 663 MACHO strong candidates in Kim et al. (2011b). From that list, only 332 objects exist in EROS-2 data set, and we find 191 matches between our EROS-2 candidates and those 332 objects. Fig. 6 shows some of the light curves of the quasar candidates for the EROS-2 data set.

Regarding the efficiency in the extraction of the CAR(1) features and the time series features, we implemented parallel processing in order to perform feature extraction and classification in a reasonable amount of time. EROS-2 and MACHO data bases are stored as a set of thousands of folders where each folder contains thousands of light curves of a given field. The feature extraction process runs as a set of parallel threads that run over different compressed files at the same time, extracting them and processing the light curves to get the features. Once the features are calculated, they are written into a common file related to a particular folder, so each compressed file has a corresponding data file that stores the feature values of all the light curves within the folder. After the feature extraction process, we run a classification process that runs in parallel over the thousands of data feature files calculated in the previous step.

5.2 MACHO data set

MACHO was a survey which observed the sky starting in 1992 July and ending in 1999 to detect microlensing events produced by Milky Way halo objects. Several tens of millions of stars were observed in the Large Magellanic Cloud (LMC), the Small Magellanic Cloud and the Galactic bulge (Alcock et al. 2000).

For the MACHO data set, we built a training set composed of 3969 non-variable stars, 127 Be stars, 78 Cepheids, 193 eclipsing binaries, 288 RR Lyrae, 574 microlensing, 359 long-period variables (LPVs) and 58 quasars. We get the variable stars from the list of known MACHO variable sources extracted from SIMBAD's MACHO variable catalogue[3] (Alcock 2001) and also from several other literature sources (Alcock et al. 1997a,b; Wood 2000; Keller et al. 2002; Thomas 2005). To get the non-variable stars, we randomly chose a subset of MACHO light curves from a few MACHO LMC fields and removed all the known MACHO variables from the subset.

Each light curve is described as a feature vector which contains 28 features: 14 features for band B and 14 features for band R, as described in Section 4.

Figs 7 and 8 show the training set projected on a two-variable feature space. We can see that σ_C and τ features show separations between two groups of classes: (i) non-variables, Cepheid and eclipsing binaries stars and (ii) quasars, microlensings, LPVs and Be stars. Combining $\overline{m}$ and τ, we can see a cluster of quasars, which overlaps with some of the Be stars, non-variables, microlensing and LPVs, but separates very well quasars from Cepheids, eclipsing binary stars and most of the non-variables. Projecting on σ_C and $\overline{m}$, we can see that quasars separate from LPVs, Cepheids, eclipsing binaries, most of the Be stars, most of the microlensings and most of the non-variables. The biggest overlap is with microlensings.

[3] http://vizier.u-strasbg.fr/viz-bin/VizieR?-source=II/247

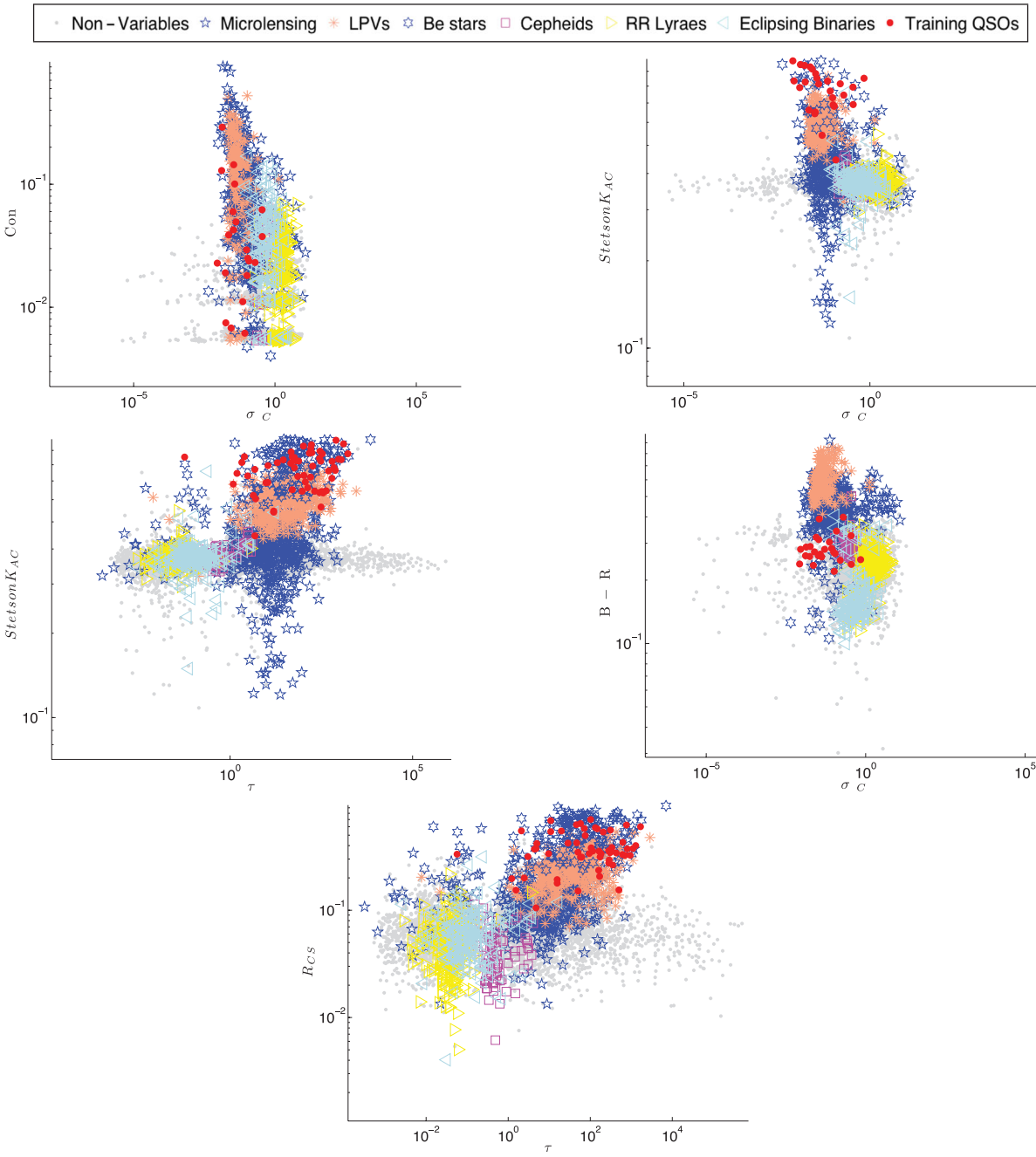

Figure 8. Projections on different pairs of features, combining CAR(1) features with time series features for MACHO training data.

By examining these projections, we can see that quasars are clustered in high values of τ, with higher values compared to eclipsing binaries, Cepheids and RR Lyraes. σ_C is very good for separating quasars from non-variables, as well as from Cepheids, RR Lyraes and eclipsing binary stars. σ_C is not a good feature for separating quasars from microlensings, Be stars and LPVs, but combining σ_C with $B - R$ we get a strong separation between them.

Table 2 shows comparative results among different classification models. We included an SVM, an RF and an RF boosted with AdaBoost. In each case, the classifier is tuned with the optimal set of parameters.

After we select and fit the model to the training set, we run the whole MACHO data (about 40 million of light curves), from where we get 2551 quasar candidates. We cross-match our candidates with the 2566 and 663 strong candidates in our previous work (Kim et al. 2011b), getting 1148 and 494 matches, respectively.

Fig. 9 shows some of the new candidates we find that are not in the previous list for MACHO candidates in Kim et al. (2011b).

There are some cases where the model confuses a periodic star with a quasar. Fig. 10 shows one example of this case.

To analyse the distribution of predicted quasars in the feature space, we show some projections of the training data plus the predicted quasars. Figs 11 and 12 show the distribution of predicted quasars, training quasars and all the other classes of stars. As in the EROS-2 case, we can see that in many cases the predicted quasars show similar distributions compared with training quasars.

Table 2. F-score for the MACHO training set using 10-fold cross-validation for different classification models. Each classifier is tuned with the optimal set of parameters. We can see that the boosted version of RF with CAR features outperforms other classification models. In all cases, using CAR features improves the result of the corresponding classifier.

SVM No CAR	SVM CAR	RF No CAR	RF CAR	AB+RF No CAR	AB+RF CAR
0.787	0.824	0.826	0.841	0.844	0.877

There are some cases where a big portion of the predicted quasars is expanded out of the concentrated cluster of training quasars, for example, combining σ_C and $B - R$.

6 SUMMARY

In this work, we present a new list of candidate quasars from MACHO and EROS-2 data sets. This new list is obtained using a new model that uses continuous autocorrelation features plus time series features to feed a boosted version of the RF classifier (Breiman 2001). With this model, we obtain a list of 1160 candidates for the EROS-2 and 2551 candidates for the MACHO data set. From our MACHO candidates, we cross-match them with the old list of candidates from Kim et al. (2011b) and we get 1148 matches. We also cross-match our EROS-2 candidates with the list of 663 MACHO strong candidates in Kim et al. (2011b). From that list, only 332 objects exist in the EROS-2 data set, and we find 131 matches

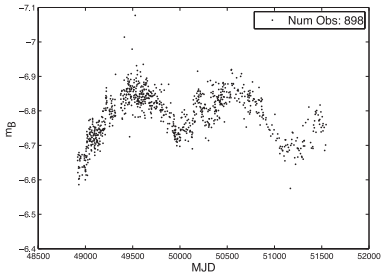

Figure 10. Light curve of a wrongly predicted quasar in MACHO data set.

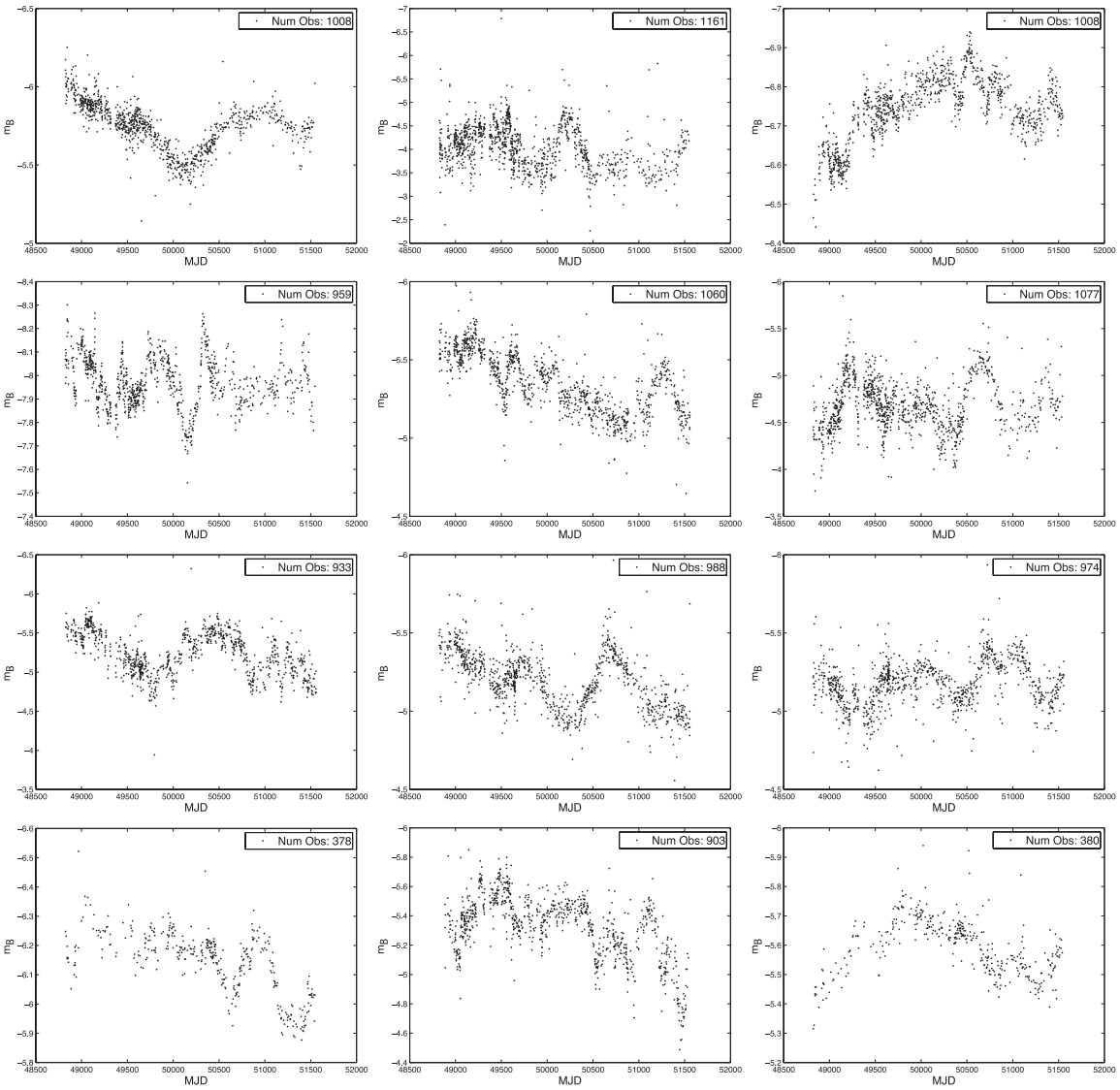

Figure 9. Light curves of new quasar candidates predicted from MACHO data set.

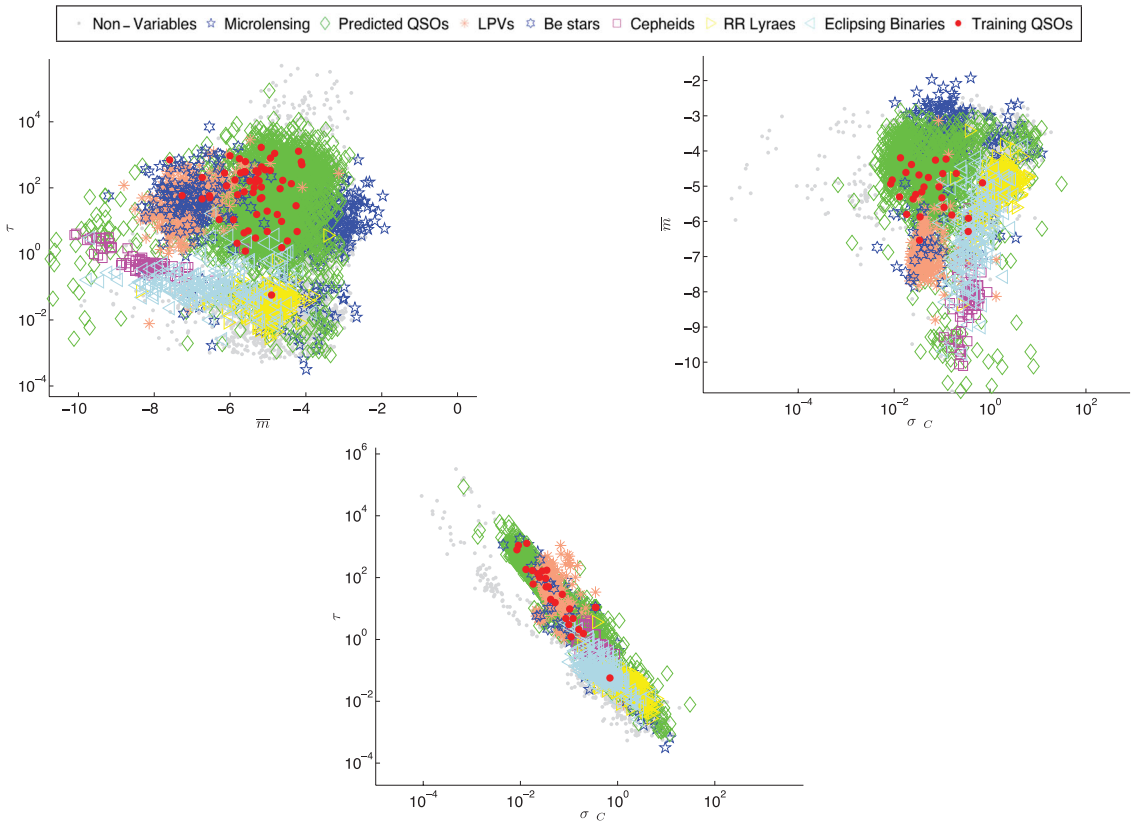

Figure 11. Predicted quasar and training star distributions projected on different pairs of CAR(1) features for MACHO data.

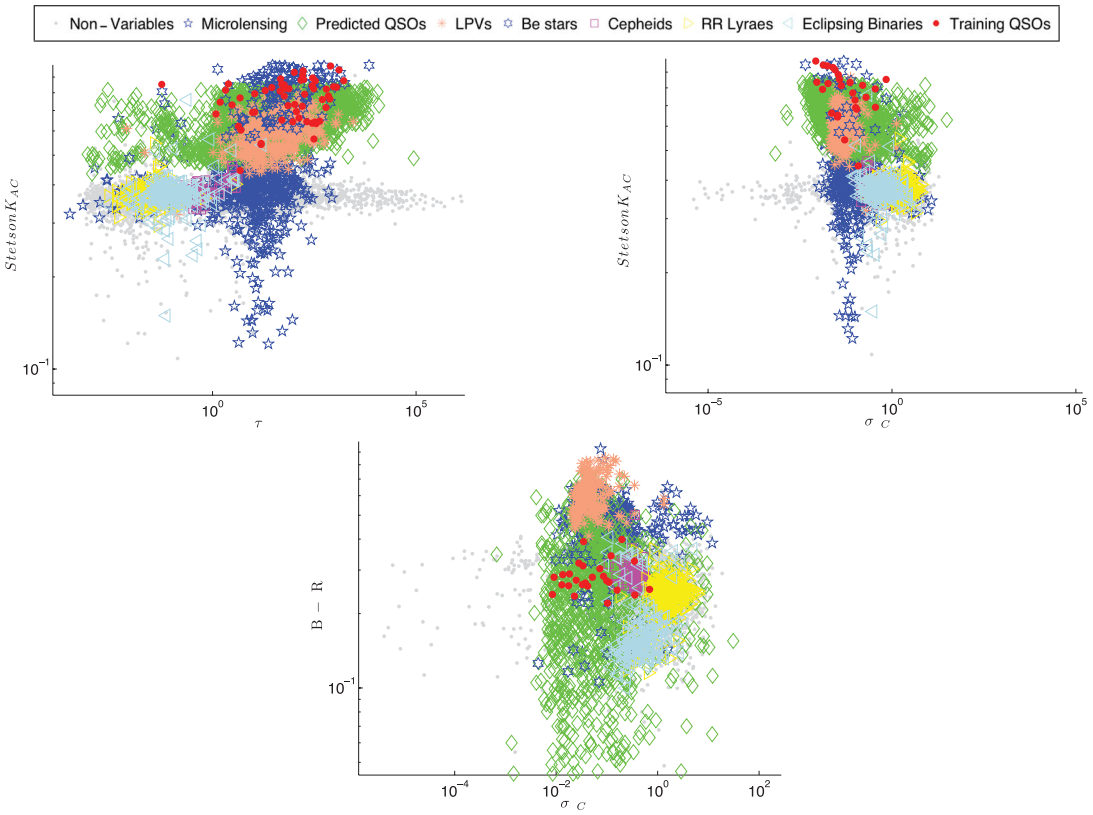

Figure 12. Predicted quasar and training star distributions for σ_C and τ features combined with three time series features for MACHO data.

Table 3. Table summarizing cross-matching results between different lists of quasars candidates.

Previous candidates MACHO (M_1) 2566	Previous strong candidates MACHO (M_2) 663	New list of MACHO candidates (M_3) 2551	List of EROS-2 candidates (E_1) 1160
Matches between (M_3) and (M_1) 1148	Matches between (M_3) and (M_2) 491	Objects from (M_2) catalogued in EROS-2 332 (*ME*)	Matches between (*ME*) and (E_1) 131

between our EROS-2 candidates and those 332 objects (see Table 3). We prove that by using boosted RF with CAR(1) features we improve the fitting of the model to the training set in both EROS-2 and MACHO data sets.

We show that quasars are well separated from many other kinds of variable stars using CAR(1) features combined with time series features. We also proved that adding CAR(1) features, SVM, RF and boosted RF improves their training accuracy. There are some challenges to overcome in future work such as the confusion of some periodic stars with quasars. We note that about 25 per cent of false positives correspond to periodic stars. We believe that by adding a dedicated module to filter periodic stars we can improve the results.

ACKNOWLEDGMENTS

This paper utilizes public domain data obtained by the MACHO project, jointly funded by the US Department of Energy through the University of California, Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48, by the National Science Foundation through the Center for Particle Astrophysics of the University of California under cooperative agreement AST-8809616 and by the Mount Stromlo and Siding Spring Observatory, part of the Australian National University. The analysis in this paper has been done using the Odyssey cluster supported by the FAS Research Computing Group at Harvard. This research has made use of the SIMBAD data base, operated at CDS, Strasbourg, France.

We thank everyone from the EROS-2 collaboration for the access granted to the data base. The EROS-2 project was funded by the CEA and the CNRS through the IN2P3 and INSU institutes.

REFERENCES

Alcock C., 2001, Variable Stars in the Large Magellanic Clouds, VizieR Online Data Catalog (http://vizier.u-strasbg.fr/viz-bin/VizieR?-source=II/247)
Alcock C. et al., 1997a, ApJ, 491, L11
Alcock C. et al., 1997b, ApJ, 479, 119
Alcock C. et al., 2000, ApJ, 542, 281
Ansari R., 1996, Vistas Astron., 40, 519
Beaulieu J. P. et al., 1996, Sci, 272, 995
Belcher J., Hampton J. S., Wilson G. T., 1994, J. R. Stat. Soc. Ser. B (Methodological), 56, 141
Bloom J. S., Richards J. W., 2011, preprint (arXiv:1104.3142)
Bloom J. S. et al., 2011, PASP, submitted (arXiv:1106.5491)
Bower R. G., Benson A. J., Malbon R., Helly J. C., Frenk C. S., Baugh C. M., Cole S., Lacey C. G., 2006, MNRAS, 370, 645
Breiman L., 1996, Machine Learning, 24, 123
Breiman L., 2001, Machine Learning, 45, 5
Brockwell P., Davis R., 2002, Introduction to Time Series and Forecasting. Springer, New York
Carliles S., Budavri T., Heinis S., Priebe C., Szalay A., 2010, ApJ, 712, 511
Cortes C., Vapnik V., 1995, Machine Learning, 20, 273
Debosscher J., Sarro L., Aerts C., Cuypers J., Vandenbussche B., Garrido R., Solano E., 2007, A&A, 475, 1159
Derue F. et al., 2002, A&A, 389, 149
Dietterich T., 1995, ACM Comput. Surv., 27, 326
Dietterich T., 2000, in Kittler J., Roli F., eds, Lecture Notes in Computer Science Vol. 1857, Proceeding First International Workshop on Multiple Classifier Systems. Springer-Verlag, London, p. 1
Duda R., Hart P., 1973, Pattern Classification and Scene Analysis. Wiley, New York
Ellaway P., 1978, Electroencephalography Clinical Neurophysiology, 45, 302
Freund Y., Schapire R., 1997, J. Comput. Syst. Sci., 170, 29
Hamadache C., 2004, PhD thesis, Univ. Louis Pasteur
Heckman T. M., Kaumann G., Brinchmann J., Charlot S., Tremonti C., White S. D. M., 2004, ApJ, 613, 109
Ivezic Z. et al., 2008, preprint (arXiv:0805.2366)
Jordan M. I., 1994, Neural Comput., 6, 181
Kaiser N. et al., 2002, in Tyson J. A., Wolff S., eds, Proc. SPIE Conf. Ser. Vol. 4836, Survey and Other Telescope Technologies and Discoveries. SPIE, Bellingham, WA, p. 154
Keller S. C., Bessell M. S., Cook K. H., Geha M., Syphers D., 2002, AJ, 124, 2039
Keller S. C. et al., 2007, Publ. Astron. Soc. Aust., 24, 1
Kelly B. C., Bechtold J., Siemiginowska A., 2009, ApJ, 698, 895
Kim D.-W., Protopapas P., Byun Y. I., Alcock C., Khardon R., Trichas M., 2011a, 735
Kim D.-W., Protopapas P., Byun Y. I., Alcock C., Khardon R., Trichas M., 2011b, ApJ, 735, 68
Kim D.-W., Protopapas P., Trichas M., Rowan-Robinson M., Khardon R., Alcock C., Byun Y. I., 2012, ApJ, 747, 107
Lomb N. R., 1976, Ap&SS, 39, 447
Metropolis N., Rosenbluth A., Rosenbluth M., Teller A., Teller E., 1953, J. Chem. Phys., 21, 1087
Nelder J., Mead R., 1965, Comput. J., 7, 308
Plamondon R., Srihari S., 2000, IEEE Trans. Pattern Anal. Mach. Intell., 22, 63
Quinlan J., 1993, C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA
Rau A. et al., 2009, PASP, 121, 1334
Richards J. W. et al., 2011, ApJ, 733, 10
Rumelhart D., Hinton G., Williams R., 1986, Learning Internal Representations by Error Propagation. MIT Press, Cambridge, MA, p. 318
Scargle J. D., 1982, ApJ, 263, 835
Stetson P. B., 1996, PASP, 108, 851
Thomas C. L. et al., 2005, ApJ, 631, 906
Tisserand P. et al., 2007, A&A, 469, 387
Trichas M., Georgakakis A., Rowan-Robinson M., Nandra K., Clements D., Vaccari M., 2009, MNRAS, 399, 663
Trichas M. et al., 2010, MNRAS, 405, 2243

Wachman G., Khardon R., Protopapas P., Alcock C., 2009, in Buntine W., Grobelnik M., Mladenic D., Shawe-Taylor J., eds, Lecture Notes in Computer Science Vol. 5782, Machine Learning and Knowledge Discovery in Databases. Springer, Berlin, p. 489

Wang Y., Khardon R., Protopapas P., 2010, in Balcázar J. L., Bonchi F., Gionis A., Sebag M., eds, Lecture Notes in Computer Science Vol. 6323, Machine Learning and Knowledge Discovery in Databases. Springer, Berlin, p. 418

Wood P. R., 2000, Publ. Astron. Soc. Aust., 17, 18

Xu L., Krzyzak A., Suen C., 1992, IEEE Trans. Syst. Man Cybern., 22, 418

Zhao W., Chellappa R., Phillips P. J., Rosenfeld A., 2003, ACM Comput. Surv., 35, 399

This paper has been typeset from a TEX/LATEX file prepared by the author.