Light Curve Analysis From Kepler Spacecraft Collected Data

Eduardo Nigri
School of Computer Science
University of St Andrews
North Haugh
St Andrews, United Kingdom KY16 9SX
eduardonigri@dcc.ufmg.br

Ognjen Arandjelović
School of Computer Science
University of St Andrews
North Haugh
St Andrews, United Kingdom KY16 9SX
ognjen.arandjelovic@gmail.com

ABSTRACT

Although scarce, previous work on the application of machine learning and data mining techniques on large corpora of astronomical data has produced promising results. For example, on the task of detecting so-called *Kepler objects of interest* (KOIs), a range of different 'off the shelf' classifiers has demonstrated outstanding performance. These rather preliminary research efforts motivate further exploration of this data domain. In the present work we focus on the analysis of *threshold crossing events* (TCEs) extracted from photometric data acquired by the Kepler spacecraft. We show that the task of classifying TCEs as being effected by actual planetary transits as opposed to confounding astrophysical phenomena is significantly more challenging than that of KOI detection, with different classifiers exhibiting vastly different performances. Nevertheless, the best performing classifier type, the random forest, achieved excellent accuracy, correctly predicting in approximately 96% of the cases. Our results and analysis should illuminate further efforts into the development of more sophisticated, automatic techniques, and encourage additional work in the area.

CCS CONCEPTS

•**Applied computing** → **Physical sciences and engineering;** •**Computing methodologies** → *Artificial intelligence;* Machine learning;

KEYWORDS

Astronomy; Big Data; photometry; space; pattern recognition; random forests; support vector machines

ACM Reference format:
Eduardo Nigri and Ognjen Arandjelović. 2017. Light Curve Analysis From Kepler Spacecraft Collected Data. In *Proceedings of ICMR '17, Bucharest, Romania, June 06-09, 2017,* 6 pages.
DOI: http://dx.doi.org/10.1145/3078971.3080544

1 INTRODUCTION

Technological advances seen in recent years have had a profound effect on the shape of applied computing. Owing to improvements in hardware and the possibility to acquire [14], store [32], and transmit [20] large amounts of information cheaply, there has been a dramatic increase in the availability of highly heterogeneous data. Acting as a part of a positive feedback loop, the broad field of artificial intelligence has seen major breakthroughs and conceptual leaps. Machine learning, pattern recognition, data science, and data mining are just some of the sub-disciplines of artificial intelligence which have come into prominence in the age of so-called Big Data [33]. The wealth of available data is an opportunity for the development of data driven (and hence evidence driven) algorithms relying on minimal hand-crafting, which have the potential to perform in a manner free of various forms of bias that humans are prone to [10].

Unsurprisingly, much of the applied research attention has focused on domains which have tangible commercial benefit or which emotionally engage the general public. Personalized product recommendations [1] typify the former. A range of applications which fall under the broad umbrella of 'social administration', have also attracted significant efforts e.g. the use of social media to track and learn about different types of emergencies [15]. Public health monitoring is also an area of great interest both to governments and individuals [7, 26].

A major application domain of interest to modern artificial intelligence and computing in general is that of scientific research. Indeed, a great and increasing amount of science now relies on the analysis of large quantities of data [2, 6, 35]. Significant efforts in the realm of personalized medicine, for example in the analysis of large scale electronic health records [3, 30, 37] have already demonstrated highly promising results.. The highly multi-modal nature of such data [5] which may consist of 'conventional' or infrared images, depth information, physical measurements of different types, demographic information, and numerous other forms, as well as the domain specific semantic gap interlaced with the interpretation of the aforementioned information, all also present major research challenges. Notwithstanding the breadth of efforts touched upon above, there are many scientific areas in which the use of state of the art artificial intelligence remains little explored, arguably in no small part because they are (often incorrectly) seen as having limited practical relevance. Yet these disciplines often stand to gain enormously from the use of data science. Astronomy is but one of them. Indeed, astronomy has over time increasingly become driven by the analysis of vast amounts of data. Data collection efforts in the form of sky surveys and others, routinely collect astonishing amounts of data. At the very least for practical reasons this collection has to be accompanied with the development of sophisticated machine learning based algorithms capable of discarding irrelevant information, automatically searching (data mining) for new information, detecting data of interest etc. To date, efforts

ICMR '17, Bucharest, Romania

978-1-4503-4701-3/17/06…$15.00
DOI: http://dx.doi.org/10.1145/3078971.3080544

towards this goal have been limited and only the most elementary techniques evaluated in pilot style experiments [16, 25, 28, 31]. Our goal in the present paper is twofold: (i) to verify independently the results reported in the existing literature, and (ii) to contribute to the understanding of the problem by comparing a greater number of classifiers than previous work.

2 TECHNICAL DETAIL

In this section we explain the types of features extracted from raw data collected by the Kepler mission, and the classification methodologies pursued in the experiments described in the present paper.

2.1 Background context

The Kepler mission was conceived by NASA to detect Earth like planets orbiting Sun like stars in the Milky Way galaxy [11]. One of the main goals of the mission is to find and determine the frequency of planets outside of the solar system (so-called exoplanets) in the habitable zone of their host stars. Such exoplanets would have temperatures that would allow liquid water to exist on their surface, which is one of the key necessary elements for making them suitable for life as we know it.

The Kepler spacecraft was launched in 2009 in an Earth-trailing heliocentric orbit. The single instrument carried by Kepler is a photometer which measures the brightness of the stars in its 115 deg^2 field-of-view [19]. Observations were sent to Earth on a monthly basis and grouped by quarters. Kepler observed a patch of the sky in the constellations of Cygnus and Lyra from May 2009 to May 2013. After losing a second reaction wheel in 2013, the spacecraft was re-purposed for the K2 mission [22].

The Kepler mission uses transit photometry to find exoplanets. As illustrated conceptually in Figure 1(a), when a planet transits in front of its host star, it blocks some of the light emitted by the star in the direction of the observer. This dip in brightness can be measured, and a periodicity in the observed dips serves as an indication of the existence of an exoplanet.

2.2 Input data and its pre-processing

As already noted in the previous section, the sole instrument onboard Kepler is a photometer – a camera, in effect – which directly senses incoming light brightness [23]. This raw data is then processed through a series of steps in order to extract features used in our experiments. Each of the steps in the pipeline will be described in more detail in Section 2.2.1. In broad terms, following the calibration of measurements a series of so-called *light curves* is created for each targeted star. Succinctly put, a light curve is a temporal characteristic variation in the brightness of a star. From light curves, an exoplanet can be detected by finding the associated periodic dips of brightness which correspond to the exoplanet's transit in front of the star from the point of view of Kepler's photometer. Sequences of transit like signals in the light curve are readily identified using multi-scale wavelet analysis and are referred to as *threshold crossing events* (TCEs). The pipeline is described in more detail next.

2.2.1 Data processing pipeline. Starting from raw photometric data sensed by charge-coupled device (CCD) detectors, the first step in the data processing pipeline involves pixel level calibration, as

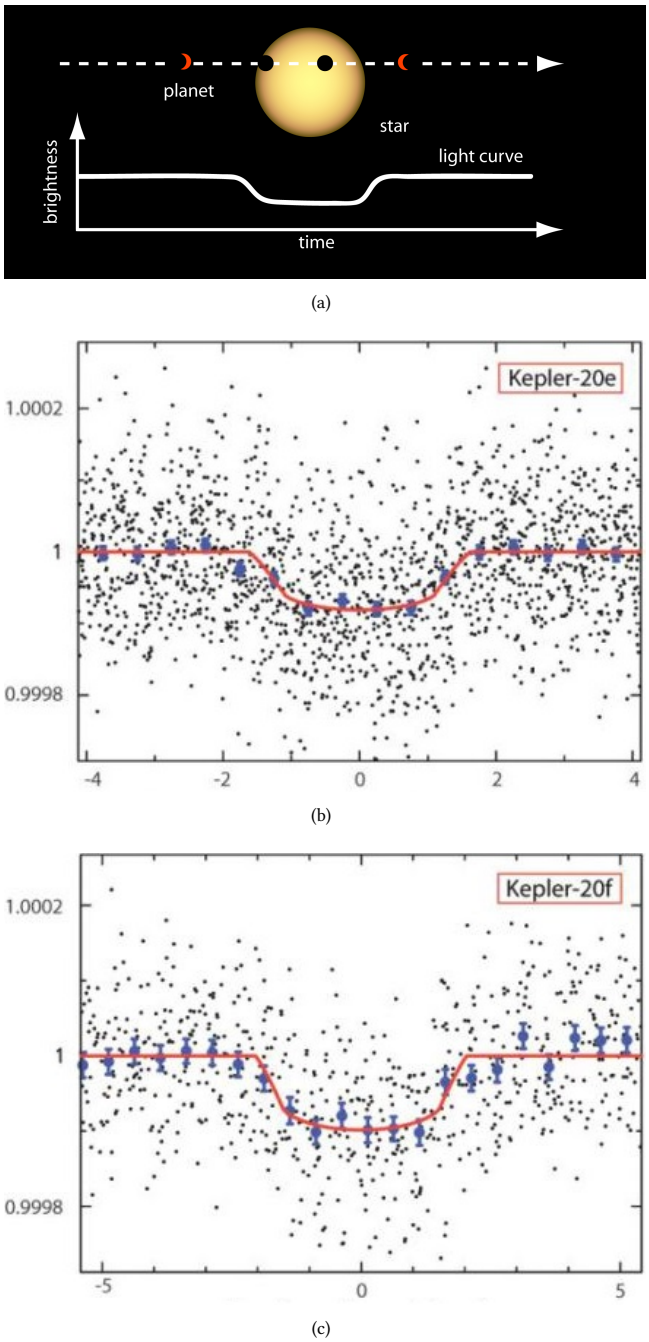

Figure 1: (a) Conceptual illustration of the extracted light curve distortion effected by a passing exoplanet. (b,c) Raw and statistically robust photometric measurements, and smooth, fitted light curves corresponding to transits of two Earth like exoplanets, Kepler-20e and Kepler-20f respectively. The horizontal axis shows time in hours relative to the time of mid-transit, and the vertical the relative flux.

shown in Figure 2. This step corrects for the effects of cosmic rays and variations in pixel sensitivity, and is performed in a standard manner for calibrating CCD data, performing corrections for bias, dark current, gain etc. Calibrated pixels are then used for photometric analysis which produces raw light curves, see Figure 1(b,c). This step too involves commonly used techniques from signal processing, such as background subtraction based on temporal averages, and robust estimation through the use of flux weighted centroids. Transiting planet detection is done next, resulting in the detection of threshold crossing events. As noted earlier, this is achieved by identifying kinks in raw light curves which also show periodicity across time. The last step in the pipeline involves what is commonly termed data validation. This is a model driven stage which results in the estimates of the relative radius of the planet, the associated period, epoch, orbit parameters, star density etc. The estimates are made by optimizing their values in a manner that fits a physical planet model.

As explained in the next section, each of the steps in the described pipeline is used for the extraction of possibly salient input features we used for KOI classification.

2.3 Extracted input features

Features used as input to the classification algorithms used in our experiments comprise the four sets used for KOI classification [31] (namely, transit fit parameters, threshold crossing event information, stellar parameters, and pixel based KOI vetting statistics – see Figure 2), TCE specific statistics, and an additional, derived feature. The derived feature was inspired by the work of McCauliff *et al.* [28] and it captures the similarity of a host star's TCEs . Its value was computed as the minimum absolute difference between periods of TCEs.

Gathering all features described above resulted in a feature set comprising 64 features in total. Two strategies were used to reduce the number of features: removing features with low variance and redundant features (showing high correlation with another feature). An empirical threshold of 1% of the mean value was used to prune features with low variance, resulting in the removal of 12 features. To measure the correlation between pairs of variables, we used the well known Pearson correlation coefficient, with values of 1 and -1 indicating perfect correlation and 0 no correlation at all. The threshold of 0.9 was adopted and the less significant feature of the pair, quantified by performing the analysis of variance, was removed. This process resulted in the final, reduced number of features of 44.

2.4 Classification methodologies

For our experiments we adopted the use of five different classification approaches. These were primarily selected on the basis of their widespread use, well understood behaviour, and promising performance in a variety of other classification tasks. Our goal was also to compare classifiers which are based on different assumptions on the relationship between different features, as well as classifiers which differ in terms of the functional forms of classification boundaries they can learn. The five compared classifiers are naïve Bayes [24], logistic regression [8], support vector machine [4], k-nearest neighbours [27], and random forest [13]. For completeness we summarize the key aspects of each next.

2.4.1 Naïve Bayes classification. Naïve Bayes classification applies the Bayes theorem by making the 'naïve' assumption of feature independence. Formally, given a set of n features $x_1, \ldots, x_n$, the associated pattern is deemed as belonging to the class y which satisfies the following condition:

$$y = \arg\max_j P(C_j) \prod_{i=1}^{n} p(x_i|C_j) \tag{1}$$

where $P(C_j)$ is the prior probability of the class C_j, and $p(x_i|C_j)$ the conditional probability of the feature x_i given class C_j (readily estimated from data using a supervised learning framework) [9].

2.4.2 Logistic regression. In logistic regression, the conditional probability of the dependent variable (class) y is modelled as a logit-transformed multiple linear regression of the explanatory variables (input features) $x_1, \ldots, x_n$:

$$P_{LR}(y = \pm 1|\mathbf{x}, \mathbf{w}) = \frac{1}{1 + e^{-y\mathbf{w}^T\mathbf{x}}}. \tag{2}$$

The model is trained (i.e. the weight parameter $\mathbf{w}$ learnt) by maximizing the likelihood of the model on the training data set, given by:

$$\prod_{i=1}^{2} Pr(y_i|\mathbf{x}_i, \mathbf{w}) = \prod_{i=1}^{2} \frac{1}{1 + e^{-y_i\mathbf{w}^T\mathbf{x}_i}}, \tag{3}$$

penalized by the complexity of the model:

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}\mathbf{w}^T\mathbf{w}}, \tag{4}$$

which can be restated as the minimization of the following regularized negative log-likelihood:

$$\mathfrak{L} = C \sum_{i=1}^{2} \log\left(1 + e^{-y_i\mathbf{w}^T\mathbf{x}_i}\right) + \mathbf{w}^T\mathbf{w}. \tag{5}$$

A coordinate descent approach described by Yu *et al.* [38] was used to minimize $\mathfrak{L}$.

2.4.3 Support vector machines. Support vector machines perform classification by constructing a series of class separating hyperplanes in a high dimensional (potentially infinitely dimensional) space into which the original input data is mapped [34]. For comprehensive detail of this regression technique the reader is referred to the original work by Vapnik [36]; herein we present a summary of the key ideas relevant to the present work.

In the context of support vector machines, the seemingly intractable task of mapping data into a very high dimensional space is achieved efficiently by performing the aforesaid mapping implicitly, rather than explicitly. This is done by employing the so-called *kernel trick* which ensures that dot products in the high dimensional space are readily computed using the variables in the original space. Given labelled training data (input vectors and the associated labels) in the form $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, a support vector machine aims to find a mapping which minimizes the number of misclassified training instances, in a regularized fashion. As mentioned earlier,

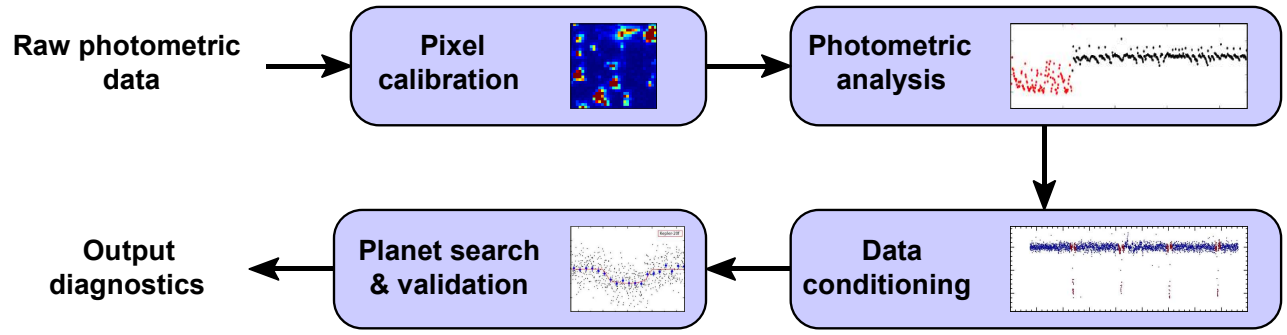

Figure 2: Feature extraction pipeline. Different sets of features, namely (1) transit fit parameters, (2) threshold crossing event information, (3) stellar parameters, and (4) pixel based KOI vetting statistics, are extracted at different stages in the pipeline.

an implicit mapping of input data $x \rightarrow \Phi(x)$ is performed by employing a Mercer-admissible kernel [29] $k(x_i, x_j)$ which allows for the dot products between mapped data to be computed in the input space: $\Phi(x_i) \cdot \Phi(x_j) = k(x_i, x_j)$. The classification vector in the transformed, high dimensional space of the form

$$w = \sum_{i=1}^{n} c_i y_i \Phi(x_i) \tag{6}$$

is sought by minimizing

$$\sum_{i=1}^{n} c_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i c_i k(x_i, x_j) y_j c_j \tag{7}$$

subject to the constraints $\sum_{i=1}^{n} c_i y_i = 0$ and $0 \leq c_i \leq 1/(2n\lambda)$. The regularizing parameter λ penalizes prediction errors.

2.4.4 k-nearest neighbours. The k-nearest neighbour classifier classifies a novel pattern comprising features $x_1, \ldots, x_n$ to the class dominant in the set of k nearest neighbours to the input pattern (in the feature space) amongst the training patterns with known class memberships [17]. The usual distance metric used is the Euclidean distance which is adopted in the present paper too.

2.4.5 Random forests. Random forest classifiers fall under the broad umbrella of ensemble based learning methods [13]. They are simple to implement, fast in operation, and have proven to be extremely successful in a variety of domains [18, 21]. The key principle underlying the random forest approach comprises the construction of many "simple" decision trees in the training stage and the majority vote (mode) across them in the classification stage. Amongst other benefits, this voting strategy has the effect of correcting for the undesirable property of decision trees to overfit training data [39]. In the training stage random forests apply the general technique known as bagging [12] to individual trees in the ensemble. Bagging repeatedly selects a random sample with replacement from the training set and fits trees to these samples. Each tree is grown without any pruning. The number of trees in the ensemble is a free parameter which is readily learnt automatically using the so-called out-of-bag error [13]; this approach is adopted in the present work as well.

3 EXPERIMENTS

In this section we describe the experiments we conducted to evaluate the effectiveness of the classification approaches described in the previous section. We examine both the effect of each classification algorithm as well as that of different features extracted from raw data.

3.1 Source data

In our experiments we adopted the same training data set used by McCauliff *et al.* [28]. TCEs are labelled as corresponding to one of three classes, namely: (i) planet candidates (PCs), (ii) , astrophysical false positives (AFPs), and (iii) non-transiting phenomena (NTP). This training set was created by matching TCEs and KOIs of the first 12 quarters of the mission, which were manually vetted by the TCERT. Detailed information is available in the DR24 catalog. In summary, the training set contains 15737 TCEs of which 3600 are PCs, 9596 AFPs, and 2541 NTPs.

3.2 Results and discussion

We started our analysis by comparing the average classification accuracies achieved by different classification approaches. The average accuracy of naïve Bayes, logistic regression, SVM, and k-nearest neighbours approaches was computed using 10-fold cross-validation. The performance of the random forest based classifier was calculated using the widely used and so-called out-of-bag error [13].

A summary of our results is shown in Table 1. As the table readily shows, the two arguably simplest approaches, namely the the naïve Bayes and logistic regression based classifiers, performed very poorly indeed, making the correct classification decision in only about 25% of the cases. The support vector machine based approach performed significantly better, achieving a respectable accuracy of 72%. The k-nearest neighbour based classifier improved on this yet further, reaching the realm of practically useful and misclassifying in only 16% of the cases (it should be noted that for k-nearest neighbour classification we optimized for the value of k on the training set and the reported results are for the learnt

Table 1: The average accuracy achieved by each of the five classifier types adopted in our experiments.

Classification methodology	Average accuracy (%)
Naïve Bayes	24.8
Logistic regression	25.1
Support vector machine	72.0
k-nearest neighbours	83.8
Random forest	95.7

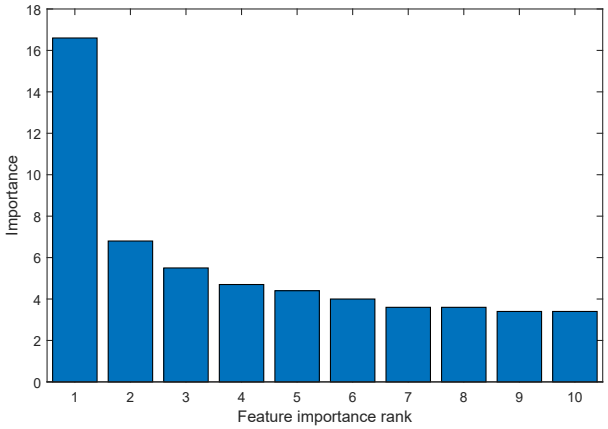

Figure 3: Relative importance of the top 10 most significant features in the context of the best performing classifier type: the random forest (see Table 1). Observe the stark dominance of the highest ranked feature, the *minimum period difference*.

optimum of $k = 4$). However, by far the best performance, and one very impressive in its own right, is that of the random forest based method which erred in only approximately 4% of the cases.

We also sought novel insight into the relative importances of different input features. Given the superior performance of the random forest based classifier we focused on this approach. A summary of our results is shown in Table 2. As the table shows, by far the most important feature was found to be the *minimum period difference*. It accounted for nearly 17% of the total importance, far exceeding the importance of the second most important feature (importance less than 7%). This feature importance distribution, illustrated further in Figure 3, is somewhat different from that previously reported on the task of KOI detection which was characterized by multiple significant features [31]. Another difference can be observed by comparing the importances of different feature types (i.e. sets), summarized in Section 2.3. While for KOI detection various transit properties accounted for half of the ten most important features, on the present task the transit, TCE, and pixel based feature sets were found to be equally represented by number in the top 10 significant features. In both cases the only top 10 ranked stellar parameter was the associated KOI count, albeit higher ranked in importance herein.

4 SUMMARY AND CONCLUSIONS

Motivated by promising results of previous research on the automation of laborious tasks in the processing of astronomical data, the present paper sought to assess the performance of several well-known classifier types in the classification of threshold crossing events detected from photometric data collected by the Kepler spacecraft. In particular, we were interested in distinguishing between events which are actually caused by planetary transits and those which are artefacts of confounding phenomena, using input features extracted from raw photometric data (images) using a multi-stage processing pipeline. Unlike on the problem of KOI classification, we found that different types of classifiers exhibited vastly different behaviours, their accuracies ranging from very low (25% for naïve Bayes) to outstandingly good (96% for random forests). This finding and our analysis should serve to encourage further work in this area. The primary focus of our future work will be on the use of raw Kepler images which would eliminate the need for the hand-crafted data pre-processing pipeline currently used to extract classifier input features.

ACKNOWLEDGEMENTS

The authors would like to thank CNPq-Brazil and the University of St Andrews for their kind support.

REFERENCES

[1] F. Abel, Q. Gao, G. J. Houben, and K. Tao. Analyzing user modeling on Twitter for personalized news recommendations. *In Proc. International Conference User Modeling, Adaptation and Personalization*, pages 1–12, 2011.
[2] V. Andrei and O. Arandjelović. Identification of promising research directions using machine learning aided medical literature analysis. *In Proc. International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2471–2474, 2016.
[3] O. Arandjelović. Discovering hospital admission patterns using models learnt from electronic hospital records. *Bioinformatics*, 31(24):3970–3976, 2015.
[4] O. Arandjelović. Learnt quasi-transitive similarity for retrieval from large collections of faces. *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 4883–4892, 2016.
[5] O. Arandjelović. Weighted linear fusion of multimodal data – a reasonable baseline? *In Proc. ACM Conference on Multimedia*, pages 851–857, 2016.
[6] A. Beykikhoshk, O. Arandjelović, D. Phung, and S. Venkatesh. Hierarchical Dirichlet process for tracking complex topical structure evolution and its application to autism research literature. *In Proc. Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 1:550–562, 2015.
[7] A. Beykikhoshk, O. Arandjelović, D. Phung, and S. Venkatesh. Overcoming data scarcity of Twitter: using tweets as bootstrap with application to autism-related topic content analysis. *In Proc. IEEE/ACM International Conference on Advances in Social Network Analysis and Mining*, pages 1354–1361, 2015.
[8] A. Beykikhoshk, O. Arandjelović, D. Phung, S. Venkatesh, and T. Caelli. Using Twitter to learn about the autism community. *Social Network Analysis and Mining*, 5(1):5–22, 2015.
[9] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, New York, USA, 2007.
[10] G. V. Bodenhausen, S. Gabriel, and M. Lineberger. Sadness and susceptibility to judgmental bias: The case of anchoring. *Psychological Science*, 11(4):320–323, 2000.
[11] W. J. Borucki, D. Koch, G. Basri, N. Batalha, and et al. Kepler planet-detection mission: introduction and first results. *Science*, 327(5968):977–980, 2010.
[12] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
[13] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
[14] M. Chan, D. Estève, C. Escriba, and E. Campo. A review of smart homes?present state and future challenges. *Computer Methods and Programs in Biomedicine*, 91(1):55–81, 2008.

Table 2: Our findings regarding the relative importance of different features in the context of a random forest classifier. The *minimum period difference* stands out as by far the most discriminative feature in the set. Legend: † median absolute deviation (MAD), pixel response function (PRF), multiple quarters (MQ), multiple event statistic (MES), Kepler Input Catalog (KIC). See http://exoplanetarchive.ipac.caltech.edu/docs/API_tce_columns.html for comprehensive information.

Feature importance rank	Feature set	Feature description	Relative importance (%)
1	–	Minimum period difference	16.6
2	1	Chi-square (χ^2) 2	6.8
3	3	Number of associated KOIs	5.5
4	2	Bootstrap false alarm probability	4.7
5	2	Weak secondary MAD-MES†	4.4
6	1	Chi-square (χ^2) 1	4.0
7	2	Weak secondary min MES	3.6
8	1	Planet-star separation	3.6
9	4	PRF MQ(KIC)†	3.4
10	4	Flux-weighted offset significance	3.4

[15] C. Chew and G. Eysenbach. Pandemics in the age of Twitter: content analysis of Tweets during the 2009 H1N1 outbreak. *PLOS ONE*, 5(11):e14118, 2010.

[16] J. L. Coughlin, F. Mullally, S. E. Thompson, J. F. Rowe, C. J. Burke, D. W. Latham, N. M. Batalha, A. Ofir, B. L. Quarles, and C. E. Henze. Planetary candidates observed by Kepler. VII. The first fully uniform catalog based on the entire 48-month data set (Q1–Q17 DR24). *The Astrophysical Journal Supplement Series*, 224(1):12, 2016.

[17] P. Cunningham and S. J. Delany. k-nearest neighbour classifiers. *Multiple Classifier Systems*, pages 1–17, 2007.

[18] D. R. Cutler, T. C. Edwards, K. H. Beard, A. Cutler, K. T. Hess, J. Gibson, and J. J. Lawler. Random forests for classification in ecology. *Ecology*, 88(11):2783–2792, 2007.

[19] M. N. Fanelli, J. M. Jenkins, S. T. Bryson, E. V. Quintana, and et al. *Kepler Data Processing Handbook*. NASA Ames Research Center, 2011.

[20] G. Fettweis and S. Alamouti. 5G: personal mobile internet beyond what cellular did to telephony. *IEEE Communications Magazine*, 52(2):140–145, 2014.

[21] P. Ghosh and B. Manjunath. Robust simultaneous registration and segmentation with sparse error reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):425–436, 2013.

[22] S. B. Howell, C. Sobeck, M. Haas, M. Still, and et al. The K2 mission: characterization and early results. *Publications of the Astronomical Society of the Pacific*, 126(938):398, 2014.

[23] J. M. Jenkins, D. A. Caldwell, H. Chandrasekaran, J. D. Twicken, and et al. Overview of the Kepler science processing pipeline. *The Astrophysical Journal Letters*, 713(2):L87, 2010.

[24] A. Jordan. On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes. *Advances in Neural Information Processing Systems*, 14:841, 2002.

[25] R. S. J. Kim, J. V. Kepner, M. Postman, M. A. Strauss, and et al. Detecting clusters of galaxies in the sloan digital sky survey I: Monte Carlo comparison of cluster detection algorithms. *The Astronomical Journal*, 123(1):20..., 2002.

[26] J. Lin and D. Ryaboy. Scaling big data mining infrastructure: the Twitter experience. *ACM SIGKDD Explorations Newsletter*, 14(2):6–19, 2013.

[27] R. Martin and O. Arandjelović. Multiple-object tracking in cluttered and crowded public spaces. *In Proc. International Symposium on Visual Computing*, 3:89–98, 2010.

[28] S. D. McCauliff, J. M. Jenkins, J. Catanzarite, C. J. Burke, J. L. Coughlin, J. D. Twicken, P. Tenenbaum, S. Seader, J. Li, and M. Cote. Automatic classification of Kepler planetary transit candidates. *The Astrophysical Journal*, 806(1):6, 2015.

[29] J. Mercer. Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society A*, 209:415–446, 1909.

[30] P. M. Nadkarni. Drug safety surveillance using de-identified EMR and claims data: issues and challenges. *J Am Med Inform Assoc*, 17(6):671–674, 2010.

[31] E. Nigri and O. Arandjelović. Machine learning based detection of Kepler objects of interest. *In Proc. ICME Workshop on Emerging Multimedia Systems and Applications*, 2017.

[32] S. F. Oliveira, K. Fürlinger, and D. Kranzlmüller. Trends in computation, communication and storage and the consequences for data-intensive science. *In Proc. IEEE International Conference on Embedded Software and Systems & High Performance Computing and Communication*, pages 572–579, 2012.

[33] F. Provost and T. Fawcett. Data science and its relationship to big data and data-driven decision making. *Big Data*, 1(1):51–59, 2013.

[34] B. Schölkopf, A. Smola, and K. Müller. *Advances in Kernel Methods – SV Learning*, chapter Kernel principal component analysis., pages 327–352. MIT Press, Cambridge, MA, 1999.

[35] A. Szalay. Extreme data-intensive scientific computing. *Computing in Science & Engineering*, 13(6):34–41, 2011.

[36] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

[37] I. Vasiljeva and O. Arandjelović. Towards sophisticated learning from EHRs: increasing prediction specificity and accuracy using clinically meaningful risk criteria. *In Proc. International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2452–2455, 2016.

[38] H.-F. Yu, F.-L. Huang, and C.-J. Lin. Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning*, 85(1–2):41–75, 2011.

[39] B. Zadrozny and C. Elkan. Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. *In Proc. IMLS International Conference on Machine Learning*, 1:609–616, 2001.