

Applied Machine-Learning Models to Identify Spectral Sub-Types of M Dwarfs from Photometric Surveys

SIRINRAT SITHAJAN ¹ AND SUKANYA MEETHONG ¹

¹*National Astronomical Research Institute of Thailand
Don Kaeo, Mae Rim, Chiang Mai, 50180, Thailand*

ABSTRACT

M dwarfs are the most abundant stars in the Solar Neighborhood and they are prime targets for searching for rocky planets in habitable zones. Consequently, a detailed characterization of these stars is in demand. The spectral sub-type is one of the parameters that is used for the characterization and it is traditionally derived from the observed spectra. However, obtaining the spectra of M dwarfs is expensive in terms of observation time and resources due to their intrinsic faintness. We study the performance of four machine-learning (ML) models—K-Nearest Neighbor (KNN), Random Forest (RF), Probabilistic Random Forest (PRF), and Multilayer Perceptron (MLP)—in identifying the spectral sub-types of M dwarfs at a grand scale by deploying broadband photometry in the optical and near-infrared. We trained the ML models by using the spectroscopically identified M dwarfs from the Sloan Digital Sky Survey (SDSS) Data Release (DR) 7, together with their photometric colors that were derived from the SDSS, Two-Micron All-Sky Survey (2MASS), and Wide-field Infrared Survey Explorer (WISE). We found that the RF, PRF, and MLP give a comparable prediction accuracy, 74%, while the KNN provides slightly lower accuracy, 71%. We also found that these models can predict the spectral sub-type of M dwarfs with $\sim 99\%$ accuracy within ± 1 sub-type. The five most useful features for the prediction are $r-z$, $r-i$, $r-J$, $r-H$, and $g-z$, and hence lacking data in all SDSS bands substantially reduces the prediction accuracy. However, we can achieve an accuracy of over 70% when the r and i magnitudes are available. Since the stars in this study are nearby ($d \lesssim 1300$ pc for 95% of the stars), the dust extinction can reduce the prediction accuracy by only 3%. Finally, we used our optimized RF models to predict the spectral sub-types of M dwarfs from the Catalog of Cool Dwarf Targets for the Transiting Exoplanet Survey Satellite (TESS), and we provide the optimized RF models for public use.

1. INTRODUCTION

M dwarfs are the major population in the Galaxy (Henry et al. 2006; Bochanski et al. 2010) and they have received attention as potential targets for detecting terrestrial-mass planets in habitable zones (Alonso-Floriano, F. J. et al. 2015; Muirhead et al. 2018). The difficulty in characterizing M dwarfs, especially when their properties are traditionally derived from spectroscopy (e.g., the accurate spectral type, surface temperature and gravity, and chemical abundance), is their intrinsic faintness. A number of studies, though not limited to M dwarfs, have tried to use the data from photometric surveys to calibrate spectroscopic stellar properties because photometric data are relatively easier and faster to obtain than spectroscopic data. For example, Mann et al. (2015, 2016) derived an empirical relation between stellar effective temperature (T_{eff}) and broadband photometry for cool stars, and Mucciarelli et al. (2021) presented an estimation of T_{eff} from Gaia (Gaia Collaboration et al. 2016) photometry.

While these studies derived a stellar property empirically, some studies have used machine-learning techniques, which recently become popular in analyzing astronomical data (see Baron 2019 and the references therein). Miller (2015) used machine learning to estimate $[\text{Fe}/\text{H}]$ to discover extremely metal-poor (EMP) stars from broadband photometry and received a $[\text{Fe}/\text{H}]$ as good as that derived from low-resolution ($R \sim 1800$) spectroscopy. Hardegree-Ullman et al. (2019) used a machine-learning technique, called the Random Forest (RF) classifier, r , J and K_s magnitudes to classify the M dwarfs that were observed by Kepler (Borucki et al. 2010) and were able to classify them more than $\sim 90\%$ accurately within one spectral sub-type of the sub-type obtained from the spectroscopic technique. Gong et al. (2022) and Gutierrez et al. (2022) also used the RF and photometric colors to distinguish ultracool dwarfs from background

contaminants and to classify them into spectral subgroups. Hardegree-Ullman et al. (2020) employed the RF classifier and RF regression to derive the spectral type, T_{eff} , surface gravity ($\log g$) and $[\text{Fe}/\text{H}]$ for ~ 200000 $K2$ stars of mid-A to mid-M type that were lacking of the spectroscopically derived parameters.

Machine learning is a branch of computer science and artificial intelligence (AI), which is broadly defined as the use of data and algorithms to imitate human intelligence, behavior, and the way that humans learn. The recent increasing availability of computing resources and large data sets, as well as the improvement in the performance and usability of algorithms implemented in machine-learning software, have led to the success of its applications in various fields, including astronomy. Specifically, supervised learning, which is an algorithm that learns from known examples where both inputs and outputs are provided, can generalize to create a predicted output for an input that it has never seen before. When it is applied to problems where the output is a category, such as stellar spectral type, it is called a classification model or a classifier.

Stellar spectral type is one of the fundamental parameters for characterizing stars. It groups together, to some degree, stars that have similar properties. Originally, the spectral type was assigned to a star based purely on the appearance of the stellar spectrum (see Gray & Corbally 2009 and reference therein). The connection between the spectral appearance and some intrinsic stellar properties, such as T_{eff} , was found later. A number of studies have assessed the relationship between the stellar spectral type and other physical properties, such as the planet occurrence rate (e.g. Fressin et al. 2013; Burke et al. 2015; Hardegree-Ullman et al. 2019).

Broadband photometric magnitude, or brightness, can be considered as a value that represents a part of stellar spectrum. The magnitudes from multiple wave bands can be thought of as an extremely low-resolution spectrum with the wavelength coverage of these wave bands. Given that the shapes of spectral energy distribution (SED) of stars with different spectral types are different, the relative values of the magnitudes from given pairs of wave bands of these stars will also be different. Therefore, photometric color, which is defined as the brightness ratio of a pair of wave bands, can be used as a feature for machine learning. By providing the colors of stars as the input (features) and the corresponding spectral types as the known output (labels) for machine learning, a classification model can be created and it can be used to predict the spectral type of a star given only input colors.

This study aims to investigate the ability of machine-learning (ML) algorithms to classify M dwarfs into spectral sub-type (M0V-M9V) by using only photometric information. The four different algorithms that we consider in this study are K-Nearest Neighbor (KNN), RF, Probabilistic Random Forest (PRF), and Multilayer Perceptron (MLP). Each of these algorithms has its own characteristics that could make it a useful model for our problem. KNN is one of the simplest ML algorithms and can be used as a baseline for our study. RF is relatively fast, interpretable, and has been widely used successfully in many applications. PRF resembles RF but is able to natively handle missing data, and take into account the uncertainty of the features and the labels. MLP, in general, has the ability to learn complex nonlinear pattern, although in this study we only explore a few basic configurations of MLP. The second purpose of this study is to find the best ML model among the four studied algorithms and to identify the most useful wavelength regions defined by well-known broadband filters in the optical and Near-Infrared (NIR) for the classification. It is important to note that when building ML models in this study, we use the labels (i.e., the spectral sub-types) that are derived from optical spectra (specifically, $\Delta\lambda \sim 3800 - 9300\text{\AA}$). For an M dwarf, the spectral sub-type derived from NIR spectra can be slightly different from that derived from the optical counterparts (e.g. Hardegree-Ullman et al. 2020). However, optical spectra have been historically used to derive spectral types of M dwarfs and the other early-type stars (Gray & Corbally 2009). Therefore, the spectral types obtained from optical spectra are widely used and consistent among stars of all classes.

In this work, we assume that the stars selected for constructing our ML models are dwarfs and are single stars, although the main catalog from which we select the stars (i.e., West et al. 2011) has 0.5% contamination and Winters et al. (2019) estimated that around one-third of M dwarfs in the Solar Neighborhood possess stellar companions. The additional steps that are described in the later sections have been performed to solidify these assumptions. The rest of this paper is structured as follows. Section 2 describes the data that we used to create the classification models. Section 3 explains the ML methods. The results are shown in Section 4. In Section 5, we use our best ML models to classify the spectral sub-types of M dwarfs in the Catalog of Cool Dwarf Targets for TESS (Muirhead et al. 2018). We describe the models that we release for public use and we explain how to use them in Section 6. A summary of this study is given in Section 7.

2. DATA

Table 1. Statistical properties of the stars used to train and evaluate ML models. The distances are geometric distances from Bailer-Jones et al. (2021). Mr is the absolute magnitude in the SDSS- r band calculated using the geometric distance and the 3D dust map of Green et al. (2018). $r-i$ is the dust-corrected $r-i$ color.

Sub-Type	Counts	Median Mr	Median $r-i$	Distance (pc) at the Percentile		
				16th	50th	84th
0	4850	7.86	0.565	519	861	1380
1	4005	8.58	0.741	433	724	1152
2	5473	9.42	0.956	325	516	842
3	6204	10.23	1.142	243	397	672
4	5001	11.13	1.338	176	314	585
5	2399	12.54	1.590	129	231	423
6	4601	13.98	1.943	129	201	291
7	5110	14.68	2.131	114	176	256
8	1358	16.33	2.617	82	121	175
9	675	17.26	2.745	63	96	150
All	39676	10.63	1.218	158	348	806

West et al. (2011), hereafter W11, provide a spectroscopic catalog of M dwarfs from the SDSS (York et al. 2000) DR7 (Abazajian et al. 2009). The catalog lists both broadband photometry and spectroscopic properties, including spectral sub-type, for 70,840 stars. Therefore, it is an ideal data set for constructing ML models. The available photometry includes SDSS $ugriz$ magnitudes and 2MASS JHK_s magnitudes. For the SDSS magnitudes, the catalog also provides dust extinction corrections, which were calculated based on Schlegel et al. (1998) 2D dust maps. The 2MASS magnitudes were obtained from the 2MASS point-source catalog (Cutri et al. 2003) by cross-matching the stars in the W11 catalog with the positions in the 2MASS catalog that lie within $5''$. Only the uniquely matched stars with high-quality 2MASS photometry were incorporated into the W11 catalog. The spectral sub-types, M0V-M9V, were derived by visual inspecting each observed SDSS spectrum ($R \sim 1800$, $\Delta\lambda \sim 3800 - 9300 \text{ \AA}$) with the Hammer spectral typing facility (Covey et al. 2007). The estimated dispersion of the classified spectral types is ~ 0.4 sub-type, which we will adopt as the error of spectral type when the ML algorithms require the value.

Given that the shape of the stellar spectrum and the relative brightness from a given pair of photometric bands are unique for each spectral type, we include W1 and W2 magnitudes from AllWISE catalog (Cutri et al. 2014), which cover $\sim 3-5 \mu\text{m}$, to increase the number of features for training the ML model and to see if they can help improve our classification models. W3 and W4 are not included because, for the majority of them, only the upper limit values of the brightness are available. We use the CDS X-Match service¹ to map the SDSS positions in the W11 catalog to the positions of stars in the AllWISE catalog that fall within $3''$. Although we adopt the $3''$ angular separation, more than 94% and 98% of the matched stars fall within $1.0''$ and $1.5''$, respectively. Because 2MASS JHK_s magnitudes are available in both W11 and AllWISE catalogs, we compare these values of the matched stars. We see that they are all consistent, and so we incorporate the AllWISE magnitudes of the matched stars to the W11 catalog.

We select stars for building ML models when the stars satisfy all of the following conditions: (1) magnitudes and the corresponding errors are available in all $ugrizJHK_s$ W1W2 bands, (2) have geometric distances from Bailer-Jones et al. (2021), (3) possess the flags in W11 catalog indicating good photometry ($q = 1$) and not white-dwarf-M-dwarf pair ($WDM = 0$), (4) have non-single-star flag = 0 in the Gaia DR3 catalog (Gaia Collaboration 2022), and (5) not in the El-Badry et al. (2021) catalog of binaries from Gaia eDR3. Although the W11 catalog indicates that its giant contamination rate is only 0.5%, we apply an additional step to enhance the purity of our sample by removing clear outliers on the $r-i$ versus Mr plane. Finally, we have 39,676 stars that satisfy the conditions and we use these stars for creating the ML models. The statistical properties of these stars and their distribution on $r-i$ versus Mr plane are shown in Table 1 and Figure 1.

3. METHODS

¹

<http://cdsxmatch.u-strasbg.fr/>

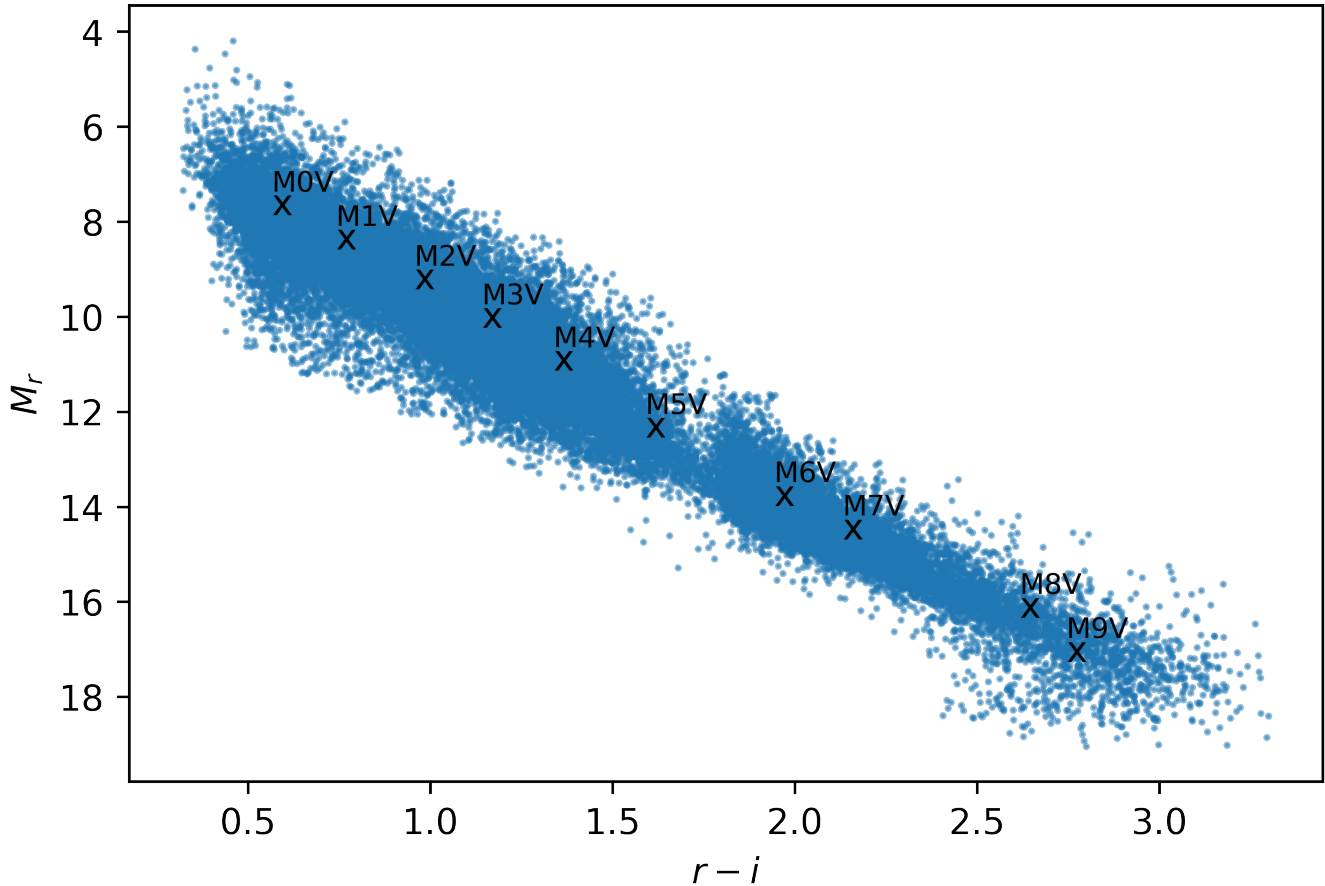


Figure 1. Color-magnitude diagram of the selected stars described in Section 2. M_r is the absolute magnitude in the SDSS- r band calculated using the geometric distance from Bailer-Jones et al. (2021) and the 3D dust map of Green et al. (2018). Note that $r-i$ is the dust-corrected $r-i$ color. The cross symbol indicates the location of the median value of the stars in each spectral sub-type on the diagram.

3.1. Training the Machine Learning Model

In supervised learning, an ML algorithm is fed with examples, each consisting of a pair of an input and a corresponding output. The algorithm then learns the pattern and finds a way to predict an output given a new input. The learning process is called *training*. The input could be a collection of properties describing the example, known as independent variables or *features*. The output is usually called the dependent variable or *label*. In this study, the features are constructed from the photometric information of the stars and the labels are the spectral sub-types of those stars.

The constructed models are expected to *generalize* to be able to predict the spectral sub-type of stars with only photometric information provided. Therefore, a subset of data needs to be excluded from the learning process and only used to evaluate the model’s performance. The subset of data used in the training is called a *training set* and the one that is used for the evaluation is called a *test set*. We split the data set described in Section 2 by stratified sampling, keeping 80% of the data for the training set and the remaining 20% for the test set.

For each star, we construct 45 photometric colors from the dust-corrected magnitudes in 10 bands, $ugrizJHK_sW1W2$. For example, color $u-r$ is obtained by the dust corrected u -band magnitude minus the dust corrected r -band magnitude. We calculate the dust extinction (A) in each band using the 3D dust map of Green et al. (2018) and the geometric distance of Bailer-Jones et al. (2021). Since Green et al. (2018) do not provide the extinction vector (R) in the u , W1, and W2 bands, we extrapolate the values from the other available bands. The corresponding error of each color is calculated using the standard error propagation formula. These photometric colors and the errors are used as features for the ML algorithms, which will be described in later sections.

Some ML algorithms, such as KNN and MLP, are sensitive to the scales of the features, so they need to be adjusted to have a similar scale. We explore two feature scaling approaches in this study. The first approach, called standardization, scales each feature so that the mean is 0 and the standard deviation is 1. The second approach, called min-max scaling, scales each feature so that the minimum value is 0 and the maximum value is 1.

When one or more classes have very low proportions when compared to the other classes, it is referred to as an imbalanced classification problem. Since most of the classification algorithms assume equally distributed classes, a severely imbalanced training set can result in a model with poorly predictive performance, especially for the minority classes (most of the contemporary works on class imbalance concentrate on imbalance ratios ranging from 1:4 up to 1:100; Krawczyk 2016). In most of the use cases, the minority class is the one that is important, and hence special techniques are needed to be applied to improve the performance of the model on the minority class. In our sample the ratios of the most minor classes M9V, M8V, and M5V to the most abundant class M3V are 1:9, 1:5, and 1:3, respectively (Table 1), and all classes are equally important. We investigate two data sampling techniques that are commonly used to mitigate the imbalanced data, (1) Random Undersampling (RUS) and (2) Synthetic Minority Oversampling Technique (SMOTE, Chawla et al. 2011), to see how they affect the overall prediction and the prediction of each class. The library *imblearn*² has these techniques implemented and the default settings are used for our study.

3.2. Machine-Learning Algorithm

In this study, we consider four different machine-learning algorithms, of which a brief overview is provided in the following subsections.

3.2.1. K-nearest Neighbors

KNN is one of the simplest machine-learning algorithms. In multi-class classification, for an unlabeled example, KNN identifies the k closest training examples in the feature space and counts how many of them belong to each class. The prediction for the unlabeled example will be the most common class. The KNN has two important hyperparameters—the number of neighbors (k) and the way the distances between the examples are measured. In this study, we use Euclidean distance which is the simplest and appropriate because all of our features are numerical variables. We use the KNN implemented in the library *scikit-learn*³.

3.2.2. Random Forest

The RF algorithm utilizes a technique called ensemble learning, in which a final prediction is made by aggregating the predictions of a group of classifiers. For RF, these classifiers are Decision Tree classifiers, and the final prediction is the class that obtains the most votes. Each of these decision trees is constructed using a different random subset of the training set (the bagging technique). When growing each tree, only a random subset of features is used to split a node. The randomness introduced by the bagging and random features reduces the variance of the overall model relative to using single Decision Tree classifiers and generally yields better performance. Some of the hyperparameters of RF are inherited from Decision Tree classifier, such as the maximum depth of each decision tree (`max_depth`) and the number of features used to split a node (`max_features`). The other hyperparameters are for controlling the ensemble, such as the total number of decision trees constituting the forest (`n_estimators`). In this study, we use the RF implemented in *scikit-learn*.

3.2.3. Probabilistic Random Forest

Probabilistic Random Forest⁴ (PRF; Reis et al. 2019) is a modified and improved version of the RF algorithm that includes the measurement uncertainty of the features or that of the label. In addition, PRF is able to handle missing data in the data set. For the uncertainty in the features, while the original RF assumes no uncertainty and an instance of data in a given node propagates either to the subsequent right or left node, the PRF uses the cumulative probability distribution function (PDF) for a particular feature to determine the probability of propagating to the left or right branch. Therefore, an instance can pass to both branches (with some probability). In principle, any function can be used for the PDF. However, in our PRF implementation, the Gaussian function is used. The node split is guided by

² <https://imbalanced-learn.org>

³ <https://scikit-learn.org>

⁴ <https://github.com/ireis/PRF>

Table 2. Accuracy of the optimized ML models built from all 45 features. The training time and prediction time are the CPU time (a laptop with a 2.3 GHz Intel Processor) used in training a model and making predictions of the stars in the test set, respectively.

Model	Accuracy (Exact Sub-Type)	Accuracy (Including ± 1 Sub-Type)	Training Time	Prediction Time
KNN	0.712 ± 0.002	0.993	0.006 s	6 s
RF	0.738 ± 0.003	0.996	1 minutes 37 s	0.3 s
PRF	0.741 ± 0.003	0.996	44 minutes 35 s	47 s
MLP	0.737 ± 0.004	0.995	5 minutes 30 s	0.6 s

the modified version of Gini impurity, which takes into account the uncertainty in the label. Once a tree is built, each leaf node would have a class probability. When making the prediction for a new instance, for each class, PRF sums the class probability from all leaf nodes weighted by the probability of that instance being in each leaf node.

3.2.4. Multilayer Perceptron

MLP is a class of the Artificial Neural Network (ANN) algorithms, where the architecture consists of one or more hidden layers. In this study, the MLP has the first hidden layer with 300 neurons and one or two additional hidden layers with 100 neurons each. Every layer is fully connected (dense). For the hidden layers, the activation function is the Rectified Linear Unit Function (ReLU). For the output layer, the activation function is softmax activation. Regarding the loss function, cross-entropy loss is used. When training, the Stochastic Gradient Descent (SGD) optimizer is used. Our hyperparameters to be optimized are the learning rate and the number of additional hidden layers (i.e., 1 or 2). Regarding regularization, a dropout of 20% is applied on hidden layers. In this study, we use the MLP implemented in *Keras*⁵.

3.3. Parameter Tuning and Evaluation Metric

Most ML algorithms have parameters that have to be specified before learning from the data. These parameters are called the model’s hyperparameters. Some important hyperparameters for each algorithm are described in Section 3.2. We find the optimized set of hyperparameters (parameter tuning) using 10-fold cross-validation and a grid search over the hyperparameter space. The option of performing scaling or resampling and which type should be used are treated as one of the hyperparameters in the tuning process. The best set of hyperparameters is selected based on the optimal accuracy, which is defined as the ratio of the correctly predicted examples to the total examples, and is used for training the final ML model. The final ML model is used to make predictions for the examples in the test set. The accuracy of the final model on the test set is used as a performance metric when we report our results and when we make a comparison of the performance of the different models.

4. RESULTS

4.1. Accuracy of the Machine-Learning Models

We compare the accuracy of the four models built from all 45 features and the optimized sets of hyperparameters. We find that the RF, PRF, and MLP models give a comparable overall prediction accuracy, which is $\sim 74\%$, and the KNN provides slightly lower accuracy, which is $\sim 71\%$. The accuracy of these models and the corresponding uncertainty are shown in Table 2. The uncertainties are the standard error of the mean test score obtained from the cross-validation. Figure 2 shows the confusion matrices of these models. For each confusion matrix, each main diagonal cell displays the fraction of correct prediction, while its left-hand and right-hand cells show the fractions of wrong predictions; that is, the matrix is normalized such that each row has the summation of one. Figure 3 shows plots of the value in the main diagonal cell, or the fraction of correct predictions, versus spectral sub-type (solid lines). We see that all models yield a similar trend. When comparing the plots to the number of stars in each spectral sub-type (dashed line), we can see some correlation (i.e., the fraction of correct prediction decreases when the number of stars used in training the ML decreases). However, they are not directly proportional. This means there are other factors beside the number of stars that affect the prediction accuracy.

⁵

<https://keras.io>

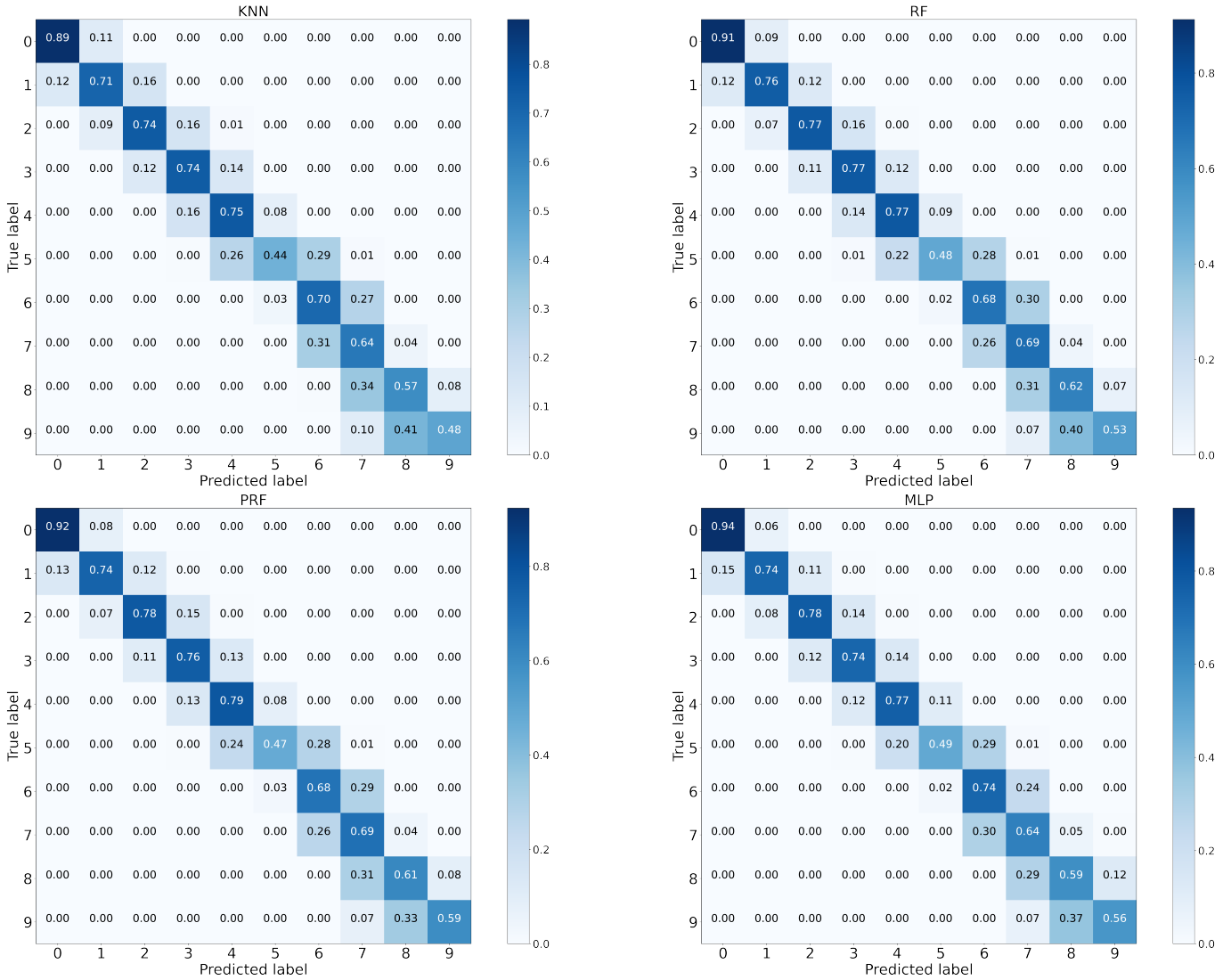


Figure 2. Normalized confusion matrices of the optimized ML models built using 45 features. Each matrix is normalized such that each row has the summation of one. The main diagonal cells indicate the correct predictions of the model for given spectral sub-types.

Better performance is expected when applying feature scaling to the KNN. However, in our case, both min-max scaling and standardization do not improve the overall accuracy. For the MLP, the standardization gives the best accuracy. Theoretically, feature scaling would have no effect for RF and PRF, and our results confirm this.

When either SMOTE or RUS is applied in the training, a significant improvement in the minority class prediction (M5V, M8V, and M9V) can be seen in all models. For the RF model, which gives the best overall accuracy among the models with the same resampling technique, the ratio of the correctly predicted stars increases by 21%, 1.2%, and 38% when SMOTE is applied and increases by 18%, 0.6%, and 44% when RUS is applied, for M5V, M8V, and M9V sub-type, respectively. However, the percentage of the increase for the sub-type does not depend on its ratio to the most abundance class. The effect of applying resampling is illustrated in Figure 4. Although there is an improvement in the minority classes predictions, with our test set that has an uneven sub-type distribution inherited from the SDSS survey (as displayed in Table 1), applying SMOTE or RUS results in a 2% decrease of the overall accuracy. We further found that, if the sub-types of the examples in the test set were uniformly distributed, the RF models with resampling would give an overall accuracy higher than the model without resampling by 3%. In other words, if the distribution of spectral sub-types in the sample to be predicted is different from that in the training sample, the overall prediction accuracy will slightly decrease. In practice, it is unlikely for there to be prior knowledge of the distribution of the

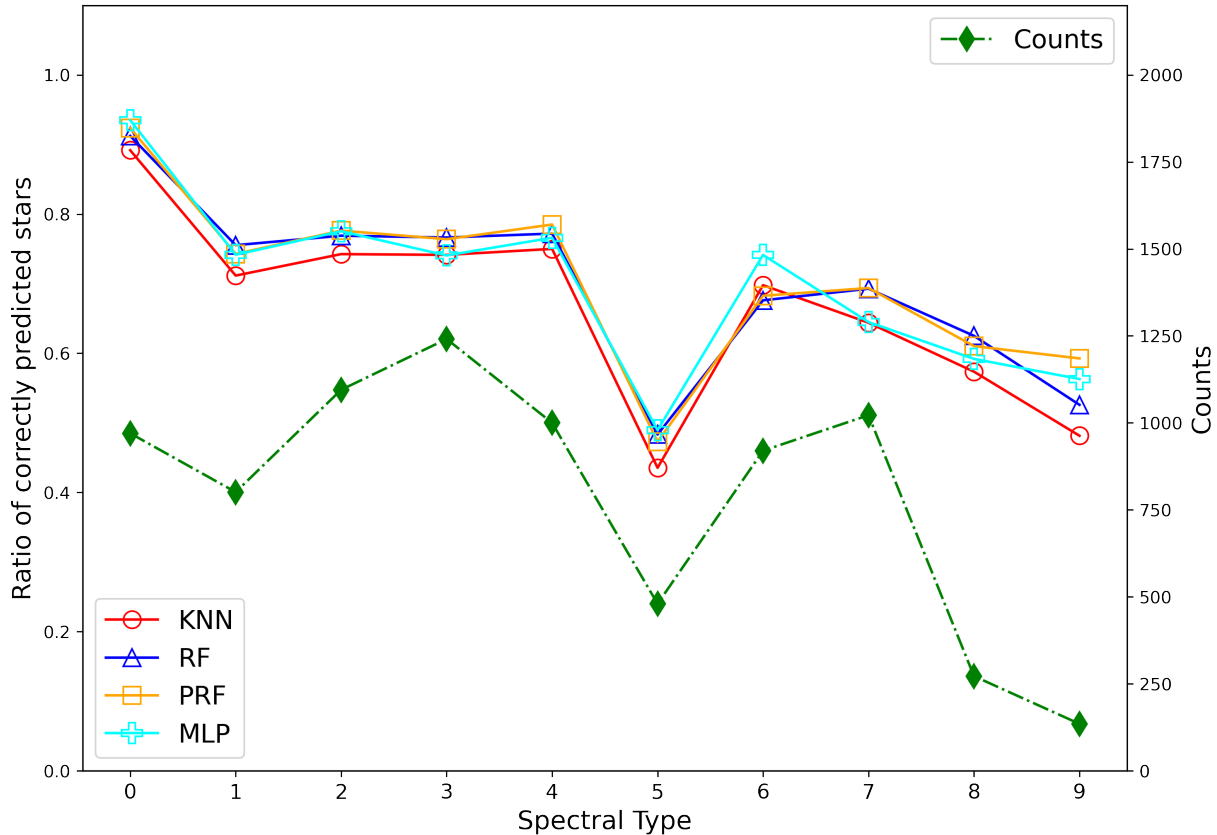


Figure 3. The ratio of correct prediction, or the value in the main diagonal cell in Figure 2, (solid line) and the number of stars (dashed line) vs. spectral sub-type. Different colors illustrate the results for different ML methods, including KNN, RF, PRF, and MLP. The lines are drawn only for clarity and no information is contained between the data points.

sub-types in the sample to be classified, and hence a slight decrease in the prediction accuracy should be expected. In this work, when we train the final models, which are selected based on the overall accuracy as described in 3.3, we do not apply any resampling unless we state this explicitly.

Generally, the finest spectral sub-type grid considered when studying a trend of astrophysical properties is 1 sub-type. We estimate the accuracy of the best models when allowing the error of ± 1 sub-type of a given sub-type and the result is shown in Table 2 and Figure 5. We see that the prediction is 99-100% correct within ± 1 sub-type for most of the spectral sub-types, and the overall accuracy is 99-100%. These results indicate that by using only broadband magnitudes in the optical and NIR and a basic ML technique, we can estimate the spectral sub-type of M dwarfs with ~ 99 -100% accuracy within 1 sub-type.

Although the RF and PRF models give a comparable result, the RF model could be built much more quickly. The PRF algorithm took into account the measurement uncertainty, including photometric error and spectral type error, explicitly when it trained the model. However, we do not see significant improvement in the accuracy when compared to the RF model. This may happen because the RF can also capture the uncertainty in the data, specifically this data set, by itself. As mentioned in Section 3.2.3, the PRF has the ability to handle missing data. We do not consider this here, but defer to Section 4.2.

4.2. Missing Features

In practice, the measured magnitudes are not always available in all 10 bands, like those used in training the models. We study how the missing measurements affects the accuracy of the models by investigating the following scenarios. First, we assume that we have the measurements from all 10 bands, but a fraction of stars have some missing values and the missing values are random. In this scenario, we can compare the performance of the PRF in handling missing values to the performance of the RF where the missing values are imputed using either mean imputation or KNN

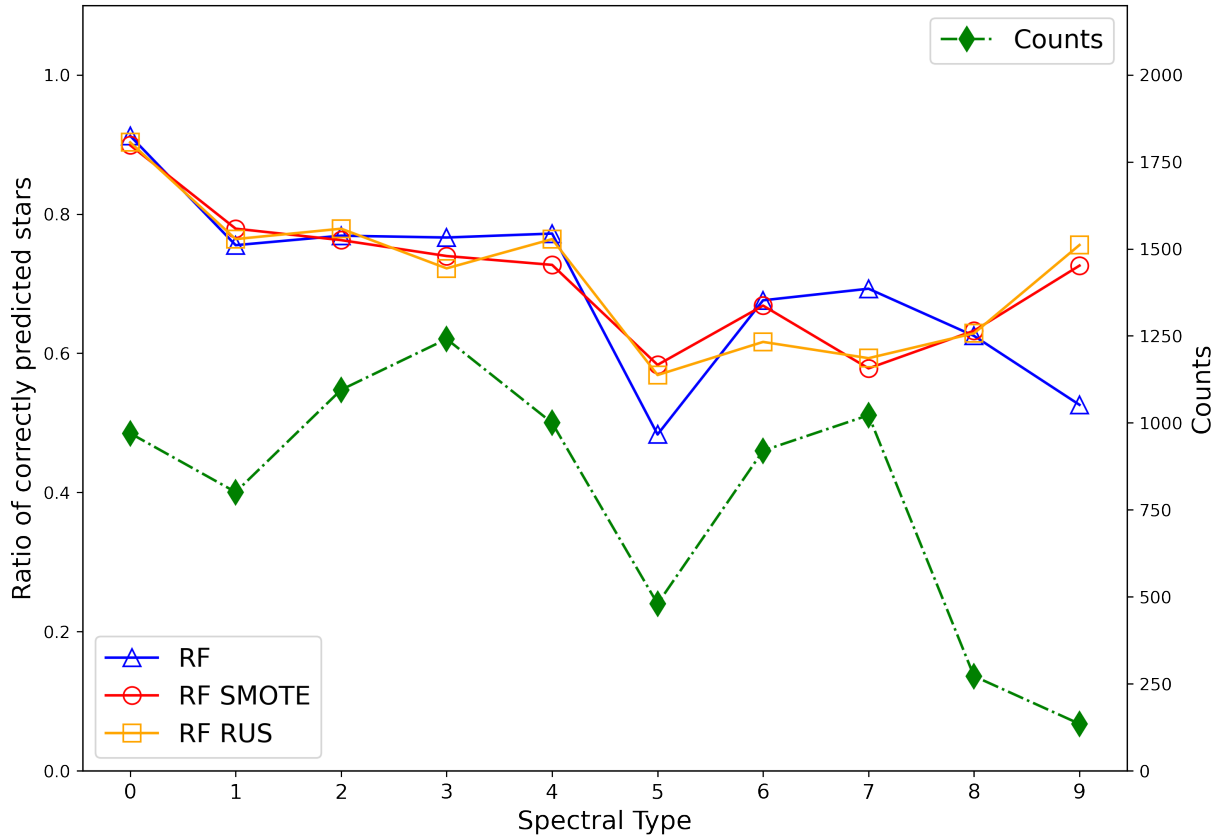


Figure 4. Comparison of the ratio of correctly predicted stars in each spectral sub-type for the RF models trained with and without resampling (solid line). The considered resampling techniques are Synthetic Minority Oversampling Technique (SMOTE) and Random Undersampling (RUS). The number of stars in each spectral sub-type is also shown (dashed line). The lines are drawn only for clarity and no information is contained between the data points.

Table 3. The accuracy of the optimized RF and PRF models (trained with 45 features) when considering the test set with different portion of missing data and different way to impute them.

% Missing Values in Each Feature	RF with Mean Impute	RF with KNN Impute	PRF
5	0.731	0.737	0.738
10	0.721	0.736	0.737
15	0.703	0.733	0.735
50	0.368	0.724	0.661

imputation. It is necessary that the features of the example to be predicted are the same as the features used to train the model. Therefore, imputation of missing features is required before making a prediction. We randomly nullify a portion of the data in the test set, as follows. We perform four experiments by varying the amount of missing data. For each feature, we randomly nullify 5%, 10%, 15%, and 50% of the data. For PRF, the missing data are handled natively. For the RF, using the optimized RF model built previously, we impute those missing data by either mean imputation or KNN imputation (with $k = 20$). We obtain the prediction accuracy that is shown in Table 3. We see that when the fraction of missing data is small, the mean imputation and KNN imputation yield a comparable accuracy. However, when the fraction of missing data is large, the KNN imputation gives much better accuracy. Comparing these results with the PRF, which is able to handle the missing data automatically, we find that the accuracy from the PRF is as good as that from the KNN imputation if the fraction of missing values is not too large (e.g. $<50\%$). Otherwise, we start to see that the KNN imputation gives a significantly better result.

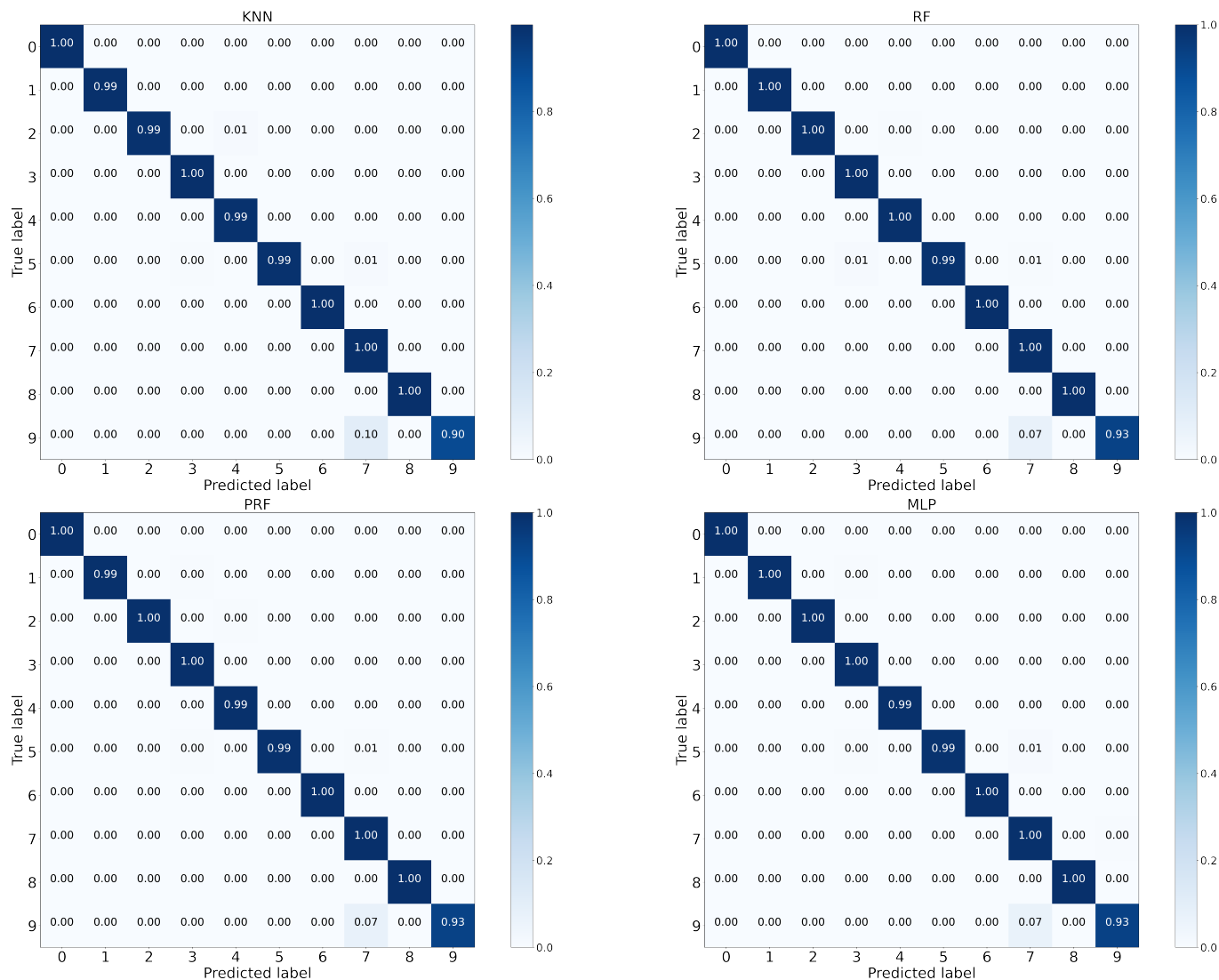


Figure 5. Normalized confusion matrices of the optimized ML models built using 45 features. Each matrix is normalized such that each row has the summation of one. The main diagonal cells indicate the correct predictions of the model for given spectral sub-types when allowing for the error of ± 1 sub-type.

With the results described in Section 4.1 and in this section, we can see that RF is one of the algorithms that provides the best accuracy. RF is also relatively fast to train and does not require feature scaling. It is also relatively easy to understand and interpret, and we can directly obtain the most important features that drive the classification results, which will be discussed later in Section 4.3. Therefore, in the remainder of Section 4, we will consider only RF as our ML model.

4.3. The Most Important Features

The importance of a feature (as implemented in *scikit-learn*⁶) in a decision tree is defined as the (normalized) weighted sum of the reduction of the impurity criterion brought by that feature on all nodes. The reduction in each node is weighted by the fraction of examples reaching that node. The importance of a feature in RF is the average of this value over all trees in the forest. We find that the five most important features are $r-z$, $r-i$, $r-J$, $r-H$, and $g-z$. We

⁶

<https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html>

Table 4. The accuracy of the optimized RF models trained with the different set of features. The first number is the accuracy of the exact prediction and the second number is the accuracy of the prediction including ± 1 sub-type error.

	All Features	Best 5 Features	<i>ugriz</i>	<i>JHK_s</i>	<i>JHK_sW1W2</i>	<i>riJHK_s</i>	<i>riJHK_sW1W2</i>
RF	0.738, 0.996	0.737, 0.994	0.733, 0.994	0.227, 0.492	0.377, 0.742	0.730, 0.993	0.731, 0.995

Table 5. The accuracy of the optimized RF models when considering different portion of missing data on the test set, which are handled by KNN imputation (k=20). The optimized RF models are trained using different set of features based on the photometric bands of the available surveys.

% Missing Values in Each Feature	All Features	<i>ugriz</i>	<i>JHK_s</i>	<i>JHK_sW1W2</i>	<i>riJHK_s</i>	<i>riJHK_sW1W2</i>
no missing values	0.738	0.733	0.227	0.377	0.730	0.731
5	0.737	0.732	0.226	0.377	0.728	0.730
10	0.736	0.731	0.224	0.377	0.728	0.730
15	0.733	0.731	0.223	0.377	0.727	0.730
50	0.724	0.717	0.199	0.367	0.714	0.725

use only these features, with re-optimization of the relevant parameters, to train a new RF model, and we receive an accuracy of 0.737, which is comparable to the best RF model that uses all 45 features (Table 4).

4.4. Features from Available Surveys

In this section, we consider the cases where the available features are defined by available photometric surveys. We consider the following cases: (1) only SDSS bands, *ugriz*; (2) only 2MASS bands *JHK_s*; and 3) 2MASS and AllWISE bands, *JHK_sW1W2*. These cases are selected based on the availability of photometric surveys rather than that of individual bandpasses. We specifically consider the case (3) because we want to see the accuracy received from using only the NIR bands without the optical bands. We do not consider pure AllWISE bands because only one feature (i.e. W1-W2) can be created. We also consider two additional cases: (4) *riJHK_s* and 5) *riJHK_sW1W2*. These cases are considered because one might want to use the *ri* magnitudes from the Pan-STARRS1 (PS1) survey (Chambers et al. 2016) when the magnitudes from the SDSS are unavailable. The *ri* bands of the PS1 are similar to those of the SDSS (Tonry et al. 2012), so they could be used when the SDSS magnitudes are unavailable.

For each of the cases described above, we train an RF model using only the available corresponding bands. We find that by using only the SDSS bands, we can achieve an accuracy that is only $\sim 1\%$ less than using all 45 bands. Meanwhile, using the 2MASS and AllWISE bands together gives $\sim 50\%$ less accuracy when compared to using only the SDSS bands. When using only the 2MASS bands, the accuracy falls to $\sim 20\%$. However, every case can provide an accuracy higher than the true randomness (10%). This indicates that photometric data can provide useful information for spectral classification. When allowing the prediction error up to ± 1 sub-type, the accuracy from using only the SDSS bands is $\sim 99\%$ and the accuracy from using only the NIR bands increases by a factor of 2 from the exact prediction. These results are presented in Table 4.

We also perform experiments similar to those in Section 4.2. That is, after training the model using only the bands available based on the available surveys, we assume that a fraction of the stars in the test set have some random missing values within those bands and the missing values are handled by KNN imputation. The results are shown in Table 5. We can see that the imputation works very well in recovering random missing values within the measured bands.

Finally, we perform an additional experiment by using the RF model trained from all 45 features. We assume we have missing features on the test set based on the availability of photometric surveys (as describe above) and we impute those missing features using KNN imputation to make each example has all 45 features. We then predict the spectral sub-type of the example. We find that the accuracy in cases (1), (2), (3), (4), and (5) decreases by 0.001, 0.064, 0.043, 0.004, and 0.007, respectively, when compared to the results in Table 4. These results indicate that it is better to have several ML models, each trained with the features matching the bands available from photometric surveys, rather than training a ML model with all 45 features and then applying imputation for the missing measurement bands, especially in cases (2) and (3). In Section 5, we demonstrate an application of this study to the data set where the spectral sub-type is unavailable.

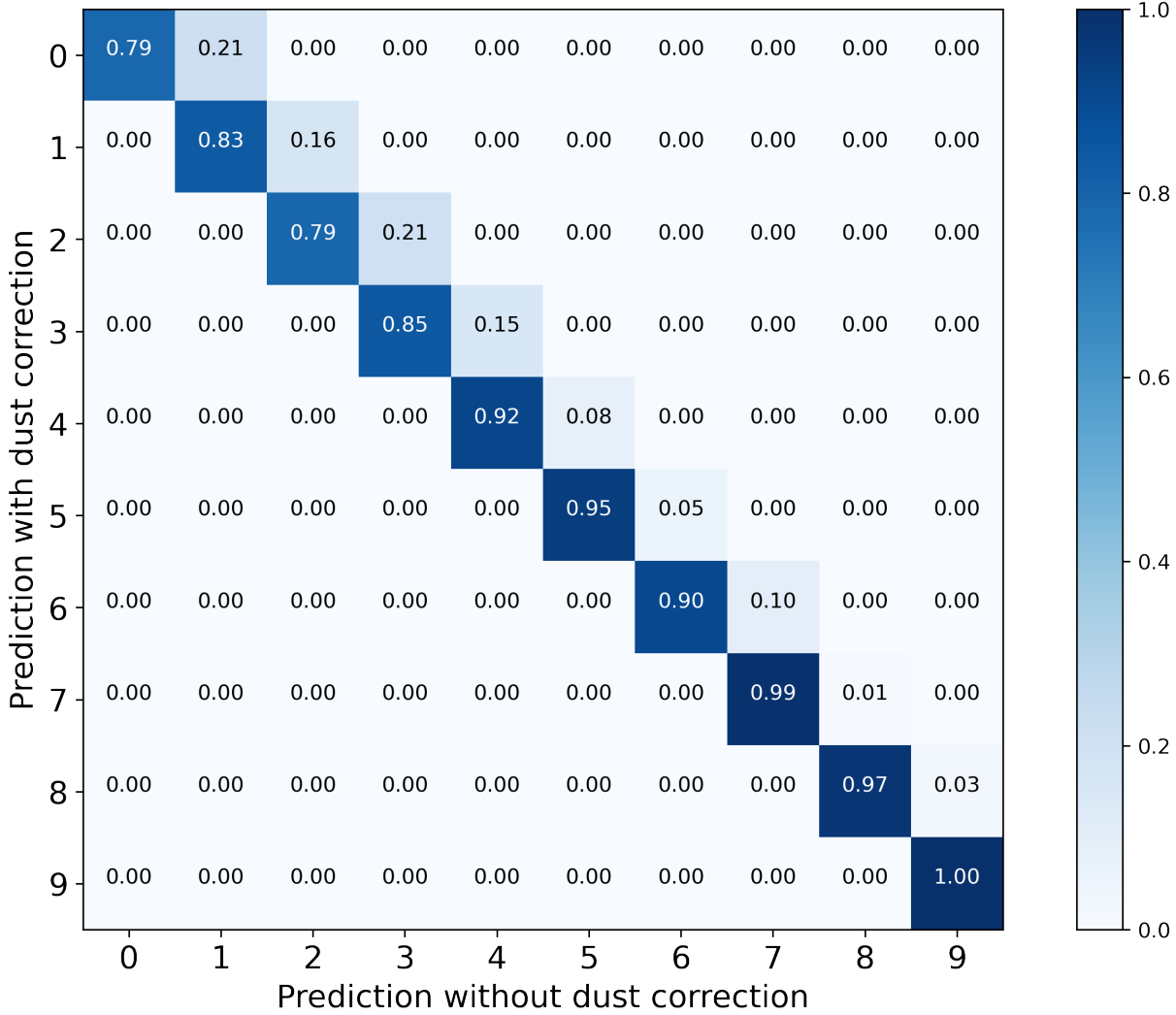


Figure 6. Modified confusion matrix comparing the predictions of examples with and without dust correction.

4.5. Effect of Dust Extinction

In this section, we investigate the effect of the dust extinction on the spectral sub-type prediction accuracy. Specifically, we determine the change in the prediction accuracy if we use the model created from dust-corrected data, like those in the previous sections, to predict an example that is not corrected for dust reddening. We use the optimized RF model created from the 45 features with dust correction to predict spectral sub-types of examples in the test set where their features are not corrected for dust extinction. We obtain an overall accuracy of 0.714, which is lower by only 3%. Figure 6 shows a modified confusion matrix where, instead of comparing the predicted spectral sub-types to the ground truth, the resulting predictions of the examples in the test set with and without dust correction are compared. We find that without dust correction, the predictions are shifted by at most +1 sub-type and the fraction of the examples that is shifted decreases with the increasing sub-type. These results are expected because without dust correction the stars appear redder and early-type stars distribute to larger distances, which are more affected by dust extinction.

Note that the stars that we use to train and test the models in this work are nearby, where 95% of the stars locate within ~ 1300 pc and their distribution can be determined from Table 1. Figure 7 illustrates the magnitudes of the extinction and photometric error in each band for the stars in this study. If a new star to be predicted has a similar or lower level of dust extinction and photometric error compared to these values, then the prediction accuracy should not be significantly worse.

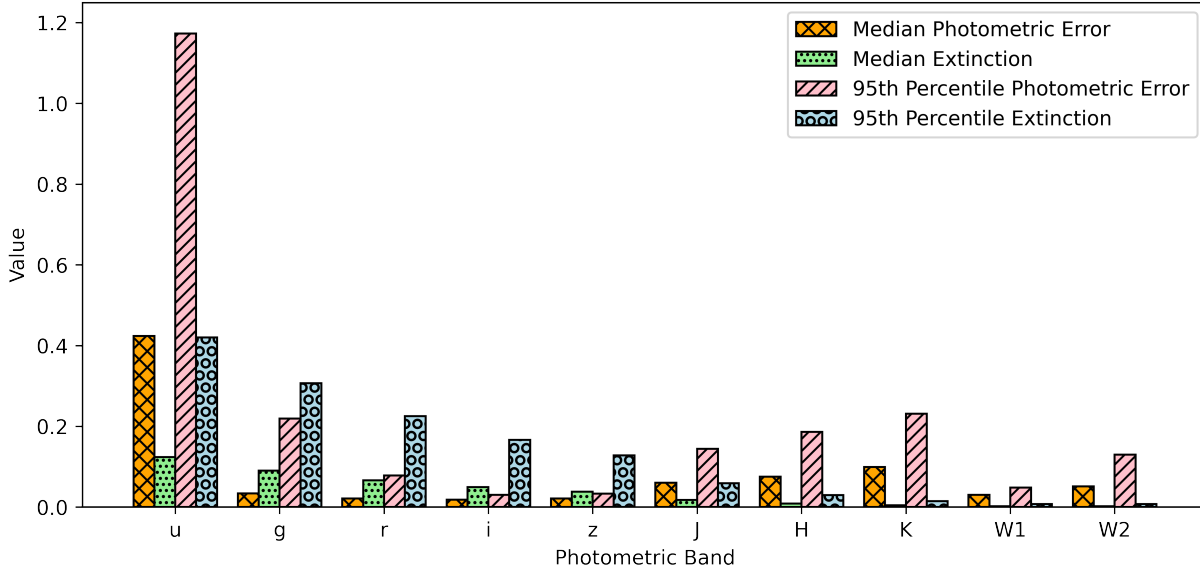


Figure 7. The median and 95th percentile values of the photometric error and dust extinction in each band for the stars that were selected in Section 2 and used to build the ML models.

5. APPLICATION

In this section, we use the optimized RF models that were created previously to predict spectral sub-types of stars in the Catalog of Cool Dwarf Targets for the Transiting Exoplanet Survey Satellite (Muirhead et al. 2018). We obtain these stars from the Exoplanet CTL v08.01 cross-matched with the TIC v8.1 catalog⁷, and only the entries with `cooldwarfs_v8` flag are selected, resulting in 3,940,291 stars in the sample hereafter CDS8. These stars are then arranged into seven groups based on the criteria described below, and their spectral sub-types are predicted using the models built corresponding to those criteria with the accuracy reported in Table 4.

Group 0 includes all stars that are in the W11 catalog. The spectral sub-types of these stars are drawn from the W11 catalog and not predicted by our models.

Group 1 includes the stars that have measured magnitudes available in all 10 bands, $ugrizJHK_sW1W2$. The predicted sub-types of these stars are 99.6% accurate within 1 sub-type of the spectroscopic sub-types.

Group 2 includes the stars that have only the SDSS $ugriz$ magnitudes. We use the model constructed using only these magnitudes to predict the spectral sub-types and the prediction accuracy within one sub-type is 99.4%.

Group 3 includes the stars that have JHK_sW1W2 magnitudes and ri magnitudes from the Pan-STARRS1 (PS1) survey, which are used as the substituting magnitudes of the unavailable SDSS ri magnitudes. We obtain the PS1 magnitudes by searching within $2''$ around a CDS8 star to find the closest PS1 star. It is possible for the closest star to actually be a different star. We estimate the percentage of the matching error by comparing the SDSS ri magnitudes of the CDS8 star to the ri magnitudes of the found PS1 star. There are around 800,000 matched stars that have the ri magnitudes from both surveys for the comparison. We find that 79% of the stars have the difference in the r magnitudes and the difference in the i magnitudes less than the 95th percentile of the magnitude errors in the corresponding bands of the stars that we used to train the models (see Figure 7). In addition, 96% of the stars have ri magnitude differences less than the 95th percentile of the dust extinction values in the corresponding bands. We have learnt previously that this level of dust extinction can reduce the prediction accuracy by only $\sim 3\%$. Therefore, we adopt the 4% as the matching error, and the prediction accuracy within 1 sub-type is estimated from $(99.5\% - 3\%) \times 96\% \sim 92.6\%$.

Group 4 includes of the stars that have JHK_s magnitudes and ri magnitudes from the PS1. The process to match the CDS8 stars to the PS1 stars is the same as that of *Group 3*, as well as the estimation of the matching error and prediction accuracy within 1 sub-type, which results in $\sim 92.4\%$.

⁷ https://archive.stsci.edu/tess/tic_ctl.html

Table 6. The number of stars in the CDS8 arranged into different groups based on the criteria explained in the text, the number of those stars that are not predicted by our ML models because their features are not in the value ranges of the corresponding features used to train the models, and the prediction accuracy corresponding to each group.

Group	All [Counts]	Not Predicted [Counts]	Expected Accuracy within ± 1 Sub-Type [%]
0	6998	6998	N/A
1	795198	59945	99.6
2	18738	5095	99.4
3	1979393	124762	92.6
4	110649	14533	92.4
5	934955	24243	74.2
6	94337	180	49.2
7	23	23	N/A
All	3940291	235779	

Group 5 includes the stars have only JHK_sW1W2 magnitudes. The prediction accuracy within 1 sub-type is 74.2%.

The stars in *Group 6* have only JHK_s magnitudes. The spectral sub-type prediction is 49.2% accurate within 1 sub-type.

Finally, *Group 7* includes the stars that do not have any photometric information in the 10 bands. Therefore, we are unable to predict the spectral sub-type of these stars.

When creating features for the prediction, we also correct for dust extinction, as we did when building the models. The CDS8 catalog provides $E(B-V)$ values for the stars locating beyond 100 pc, and there are two sources of the $E(B-V)$: Schlegel dust map and Pan-STARRS dust map. For the Schlegel dust map, we calculate the dust extinction using the extinction vector (R) from Table 6 of Schlafly & Finkbeiner (2011), assuming $R_V = 3.1$, if R is available for the given band, and we do not correct for the extinction for the band without the R value. We estimate the effect of correcting only the bands with available R (leaving the other bands uncorrected), similar to what we did in Section 4.5. We find that this affects the prediction accuracy by less than 1%. For the Pan-STARRS, the extinction vector values that we use come from Table 1 of Green et al. (2018). For the bands where the values are unavailable, we use the extrapolated values, such as those we adopted when training the models in Section 3.

Before we make a prediction for each star, we need to check whether all of the features of the star are in the value ranges of the features that we used to build the models. We do this because the ML models are unable to predict the result by extrapolation. Because the CDS8 also contains late-K dwarfs, we need to exclude them from the prediction. Table 6 summarizes the total number of stars in each group and the number of stars that are not classified by our ML models.

The results from the predictions in this section are available for downloading at our GitHub repository: <https://github.com/sirinrats/ml-md-class/>.

6. AVAILABLE MODELS

We provide the optimized RF models that we created and used in Section 5 for public use. We briefly describe the models again here for reference. The models were trained with different number of features, as summarized below:

Model 1 was trained with colors derived from all 10 bands, $ugrizJHK_sW1W2$.

Model 2 was trained with colors derived from $ugriz$ bands.

Model 3 was trained with colors derived from $riJHK_sW1W2$ bands.

Model 4 was trained with colors derived from $riJHK_s$ bands.

Model 5 was trained with colors derived from JHK_sW1W2 bands.

Model 6 was trained with colors derived from JHK_s bands.

All of the models listed above were trained without feature scaling and resampling. The hyperparameters were optimized through 10-fold cross-validation. For each model, when the required features are missing, the missing features will be imputed using the KNN imputation. The user can decide which model(s) to be used, based on the availability of the features of the stars to be predicted. For example, if W1 and W2 are missing, then the user can use the Model 1 where the features related to W1 and W2 will be imputed by the model, with the caveat that the

expected accuracy will drop (refer to Table 3) or the user can use *Model 2* with the expected accuracy as reported in Table 4.

The models listed above were trained using the examples with the spectral sub-type distribution inherited from the W11 sample (Table 1). If the user thinks that the distribution of the sample to be classified is different from the W11 sample, then they might want to use the models trained with the sample that has the sub-type uniformly distributed. Therefore, we also provide the corresponding models with Random Undersampling applied.

The models, corresponding confusion matrices, and reported accuracy are available at our GitHub repository. An example snippet code to run the model is also provided.

7. SUMMARY

We examined how well machine-learning models could classify the different spectral sub-types of M dwarfs by utilizing only their photometric magnitudes in the optical and NIR bands. The models that we considered were KNN, RF, PRF, and MLP, and the magnitudes that we used were from the SDSS-*ugriz* bands, 2MASS-*JHK_s* bands, and AllWISE-W1W2 bands. The findings indicate that all of the models that we tested were capable of correctly classifying over 70% of M dwarfs into the corresponding sub-type, as determined by spectroscopic analysis. Additionally, the models were able to classify more than 99% of the M dwarfs into sub-types within a margin of error of ± 1 sub-type when compared to the results of spectroscopic classification. The most useful wave bands for the classification were *r*, *z*, *i*, *J*, *H*, and *g* bands. Given that the stars examined in this study were relatively close to the Earth ($\lesssim 1300$ pc), the effect of dust extinction was small, degrading the classification accuracy by only 3%.

Resampling techniques such as SMOTE or Random Undersampling, which adjust the number of examples in each class to be evenly distributed, could significantly improve the predictive performance of minority classes. However, with our test set, which has an uneven distribution inherited from the SDSS survey, the overall accuracy decreased by 2%.

We employed the RF models, which demonstrated the highest level of performance among the machine-learning models that we evaluated, and classified the spectral sub-type (M0V-M9V) of 94% of stars in the Muirhead et al. (2018) catalog. Furthermore, we have made the models available for public use via our GitHub repository.

It should be noted that the concept of model performance, like accuracy, only makes sense when considering a sample of data. When using an ML model to predict a single target, the result can be either correct or incorrect. Therefore, the provided ML models should not be used for the study of an individual star but rather used in the study of a large sample.

This research aims to contribute to the development of new and innovative methods for analyzing large astronomical data sets that are difficult to manage using traditional methods. It provides insight into the level of accuracy that can be achieved under certain resources and constraints. This research could prove particularly beneficial for upcoming surveys such as the Large Synoptic Survey Telescope (LSST; Ivezić et al. (2019)), which is expected to generate an enormous amount of data that would be impossible to analyze using conventional techniques.

8. DATA ACCESS

The public data described here may be obtained from the following websites.

SDSS DR7 M Dwarfs (J/AJ/141/97/catalog) : <https://vizier.cds.unistra.fr/>

2MASS : <https://irsa.ipac.caltech.edu/Missions/2mass.html>

AllWISE : <https://wise2.ipac.caltech.edu/docs/release/allwise/>

Gaia DR3 : <https://gea.esac.esa.int/archive/>

Gaia eDR3 distances: <https://gea.esac.esa.int/archive/>

Wide Binaries from Gaia eDR3 : https://zenodo.org/record/4435257#.Y_QyLuxBzvU

TESS : <https://archive.stsci.edu/tess/tic.ctl.html>

Pan-STARRS1 : <https://catalogs.mast.stsci.edu/panstarrs/>

9. ACKNOWLEDGMENTS

The authors would like to thank Dr. Apichart Hortiangtham for insights, suggestions, and fruitful discussions on machine learning methods. The authors would like to thank the anonymous referees for their suggestions on how to enhance this manuscript.

This research was funded by the National Astronomical Research Institute of Thailand (Public Organization) (NARIT) and Thailand Science Research and Innovation (TSRI). The authors extend our sincere appreciation to NARIT and TSRI for their support.

This research made use of the NASA Astrophysics Data System (ADS) Bibliographic Services, the SIMBAD database (Wenger et al. 2000), the cross-match service provided by CDS Strasbourg, `scikit-learn` (Pedregosa et al. 2011), `NumPy` (Harris et al. 2020), `pandas` (McKinney 2010), and `Keras` (Chollet et al. 2015).

This work made extensive use of the Sloan Digital Sky Survey (SDSS) data. Funding for the SDSS and SDSS-II has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, the U.S. Department of Energy, the National Aeronautics and Space Administration, the Japanese Monbukagakusho, and the Max Planck Society, and the Higher Education Funding Council for England. The SDSS is managed by the Astrophysical Research Consortium (ARC) for the Participating Institutions. The Participating Institutions are the American Museum of Natural History, Astrophysical Institute Potsdam, University of Basel, University of Cambridge, Case Western Reserve University, The University of Chicago, Drexel University, Fermilab, the Institute for Advanced Study, the Japan Participation Group, The Johns Hopkins University, the Joint Institute for Nuclear Astrophysics, the Kavli Institute for Particle Astrophysics and Cosmology, the Korean Scientist Group, the Chinese Academy of Sciences (LAMOST), Los Alamos National Laboratory, the Max-Planck-Institute for Astronomy (MPIA), the Max-Planck-Institute for Astrophysics (MPA), New Mexico State University, Ohio State University, University of Pittsburgh, University of Portsmouth, Princeton University, the United States Naval Observatory, and the University of Washington. The SDSS Web site is <http://www.sdss.org/>.

This publication made use of data products from the Two Micron All Sky Survey (2MASS), which is a joint project of the University of Massachusetts and the Infrared Processing and Analysis Center/California Institute of Technology, funded by the National Aeronautics and Space Administration and the National Science Foundation.

This publication made use of data products from the Wide-field Infrared Survey Explorer (*WISE*), which is a joint project of the University of California, Los Angeles, and the Jet Propulsion Laboratory/California Institute of Technology, funded by the National Aeronautics and Space Administration.

This paper includes data collected with the *TESS* mission, obtained from the MAST data archive at the Space Telescope Science Institute (STScI). Funding for the *TESS* mission is provided by the NASA Explorer Program. STScI is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS 5-26555.

This publication made use of data products from the Pan-STARRS1 (PS1) Surveys. The PS1 and the PS1 public science archive have been made possible through contributions by the Institute for Astronomy, the University of Hawaii, the Pan-STARRS Project Office, the Max-Planck Society and its participating institutes, the Max Planck Institute for Astronomy, Heidelberg and the Max Planck Institute for Extraterrestrial Physics, Garching, The Johns Hopkins University, Durham University, the University of Edinburgh, the Queen's University Belfast, the Harvard-Smithsonian Center for Astrophysics, the Las Cumbres Observatory Global Telescope Network Incorporated, the National Central University of Taiwan, the Space Telescope Science Institute, the National Aeronautics and Space Administration under Grant No. NNX08AR22G issued through the Planetary Science Division of the NASA Science Mission Directorate, the National Science Foundation Grant No. AST-1238877, the University of Maryland, Eotvos Lorand University (ELTE), the Los Alamos National Laboratory, and the Gordon and Betty Moore Foundation.

This work made use of data from the European Space Agency (ESA) mission *Gaia* (<https://www.cosmos.esa.int/gaia>), processed by the *Gaia* Data Processing and Analysis Consortium (DPAC, <https://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.

REFERENCES

- Abazajian, K. N., Adelman-McCarthy, J. K., Agüeros, M. A., et al. 2009, *ApJS*, 182, 543, doi: [10.1088/0067-0049/182/2/543](https://doi.org/10.1088/0067-0049/182/2/543)
- Alonso-Floriano, F. J., Morales, J. C., Caballero, J. A., et al. 2015, *A&A*, 577, A128, doi: [10.1051/0004-6361/201525803](https://doi.org/10.1051/0004-6361/201525803)
- Bailer-Jones, C. A. L., Rybizki, J., Fouesneau, M., Demleitner, M., & Andrae, R. 2021, *VizieR Online Data Catalog*, I/352
- Baron, D. 2019, arXiv e-prints, arXiv:1904.07248. <https://arxiv.org/abs/1904.07248>

- Bochanski, J. J., Hawley, S. L., Covey, K. R., et al. 2010, *AJ*, 139, 2679, doi: [10.1088/0004-6256/139/6/2679](https://doi.org/10.1088/0004-6256/139/6/2679)
- Borucki, W. J., Koch, D., Basri, G., et al. 2010, *Science*, 327, 977, doi: [10.1126/science.1185402](https://doi.org/10.1126/science.1185402)
- Burke, C. J., Christiansen, J. L., Mullally, F., et al. 2015, *ApJ*, 809, 8, doi: [10.1088/0004-637X/809/1/8](https://doi.org/10.1088/0004-637X/809/1/8)
- Chambers, K. C., Magnier, E. A., Metcalfe, N., et al. 2016, arXiv e-prints, arXiv:1612.05560. <https://arxiv.org/abs/1612.05560>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. 2011, arXiv e-prints, arXiv:1106.1813. <https://arxiv.org/abs/1106.1813>
- Chollet, F., et al. 2015, Keras, GitHub. <https://github.com/fchollet/keras>
- Covey, K. R., Ivezić, Ž., Schlegel, D., et al. 2007, *AJ*, 134, 2398, doi: [10.1086/522052](https://doi.org/10.1086/522052)
- Cutri, R. M., Skrutskie, M. F., van Dyk, S., et al. 2003, 2MASS All Sky Catalog of point sources.
- Cutri, R. M., Wright, E. L., Conrow, T., et al. 2014, *VizieR Online Data Catalog*, II/328
- El-Badry, K., Rix, H.-W., & Heintz, T. M. 2021, *MNRAS*, 506, 2269, doi: [10.1093/mnras/stab323](https://doi.org/10.1093/mnras/stab323)
- Fressin, F., Torres, G., Charbonneau, D., et al. 2013, *ApJ*, 766, 81, doi: [10.1088/0004-637X/766/2/81](https://doi.org/10.1088/0004-637X/766/2/81)
- Gaia Collaboration. 2022, *VizieR Online Data Catalog*, I/355
- Gaia Collaboration, Prusti, T., de Bruijne, J. H. J., et al. 2016, *A&A*, 595, A1, doi: [10.1051/0004-6361/201629272](https://doi.org/10.1051/0004-6361/201629272)
- Gong, Z., Nava Vega, A., Gauna Gutierrez, E., et al. 2022, *Research Notes of the American Astronomical Society*, 6, 74, doi: [10.3847/2515-5172/ac6521](https://doi.org/10.3847/2515-5172/ac6521)
- Gray, R. O., & Corbally, Christopher, J. 2009, *Stellar Spectral Classification*
- Green, G. M., Schlafly, E. F., Finkbeiner, D., et al. 2018, *MNRAS*, 478, 651, doi: [10.1093/mnras/sty1008](https://doi.org/10.1093/mnras/sty1008)
- Gutierrez, E. G., Maytorena, A. M., Gong, Z., et al. 2022, *Research Notes of the American Astronomical Society*, 6, 75, doi: [10.3847/2515-5172/ac6522](https://doi.org/10.3847/2515-5172/ac6522)
- Hardegree-Ullman, K. K., Cushing, M. C., Muirhead, P. S., & Christiansen, J. L. 2019, *AJ*, 158, 75, doi: [10.3847/1538-3881/ab21d2](https://doi.org/10.3847/1538-3881/ab21d2)
- Hardegree-Ullman, K. K., Zink, J. K., Christiansen, J. L., et al. 2020, *ApJS*, 247, 28, doi: [10.3847/1538-4365/ab7230](https://doi.org/10.3847/1538-4365/ab7230)
- Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, *Nature*, 585, 357, doi: [10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2)
- Henry, T. J., Jao, W.-C., Subasavage, J. P., et al. 2006, *AJ*, 132, 2360, doi: [10.1086/508233](https://doi.org/10.1086/508233)
- Ivezić, Ž., Kahn, S. M., Tyson, J. A., et al. 2019, *ApJ*, 873, 111, doi: [10.3847/1538-4357/ab042c](https://doi.org/10.3847/1538-4357/ab042c)
- Krawczyk, B. 2016, *Progress in Artificial Intelligence*, 5, 221, doi: [10.1007/s13748-016-0094-0](https://doi.org/10.1007/s13748-016-0094-0)
- Mann, A. W., Feiden, G. A., Gaidos, E., Boyajian, T., & von Braun, K. 2015, *ApJ*, 804, 64, doi: [10.1088/0004-637X/804/1/64](https://doi.org/10.1088/0004-637X/804/1/64)
- . 2016, *ApJ*, 819, 87, doi: [10.3847/0004-637X/819/1/87](https://doi.org/10.3847/0004-637X/819/1/87)
- McKinney. 2010, in *Proceedings of the 9th Python in Science Conference*, ed. Stéfan van der Walt & Jarrod Millman, 56 – 61, doi: [10.25080/Majora-92bf1922-00a](https://doi.org/10.25080/Majora-92bf1922-00a)
- Miller, A. A. 2015, *ApJ*, 811, 30, doi: [10.1088/0004-637X/811/1/30](https://doi.org/10.1088/0004-637X/811/1/30)
- Mucciarelli, A., Bellazzini, M., & Massari, D. 2021, *A&A*, 653, A90, doi: [10.1051/0004-6361/202140979](https://doi.org/10.1051/0004-6361/202140979)
- Muirhead, P. S., Dressing, C. D., Mann, A. W., et al. 2018, *AJ*, 155, 180, doi: [10.3847/1538-3881/aab710](https://doi.org/10.3847/1538-3881/aab710)
- Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, *Journal of Machine Learning Research*, 12, 2825. <https://arxiv.org/abs/1201.0490>
- Reis, I., Baron, D., & Shahaf, S. 2019, *AJ*, 157, 16, doi: [10.3847/1538-3881/aaf101](https://doi.org/10.3847/1538-3881/aaf101)
- Schlafly, E. F., & Finkbeiner, D. P. 2011, *ApJ*, 737, 103, doi: [10.1088/0004-637X/737/2/103](https://doi.org/10.1088/0004-637X/737/2/103)
- Schlegel, D. J., Finkbeiner, D. P., & Davis, M. 1998, *ApJ*, 500, 525, doi: [10.1086/305772](https://doi.org/10.1086/305772)
- Tonry, J. L., Stubbs, C. W., Lykke, K. R., et al. 2012, *ApJ*, 750, 99, doi: [10.1088/0004-637X/750/2/99](https://doi.org/10.1088/0004-637X/750/2/99)
- Wenger, M., Ochsenbein, F., Egret, D., et al. 2000, *A&AS*, 143, 9, doi: [10.1051/aas:2000332](https://doi.org/10.1051/aas:2000332)
- West, A. A., Morgan, D. P., Bochanski, J. J., et al. 2011, *AJ*, 141, 97, doi: [10.1088/0004-6256/141/3/97](https://doi.org/10.1088/0004-6256/141/3/97)
- Winters, J. G., Henry, T. J., Jao, W.-C., et al. 2019, *AJ*, 157, 216, doi: [10.3847/1538-3881/ab05dc](https://doi.org/10.3847/1538-3881/ab05dc)
- York, D. G., Adelman, J., Anderson, John E., J., et al. 2000, *AJ*, 120, 1579, doi: [10.1086/301513](https://doi.org/10.1086/301513)