

**Поиск L и T коричневых карликов в данных
современных обзоров неба методами машинного
обучения**

**Курсовая работа
аспирантки
аспирантской школы по физике
НИУ ВШЭ
Авдеевой А. С.**

Оглавление

Введение	2
1. Данные	3
1.1. Генерация дополнительных признаков	5
1.2. Работа с пропущенными значениями	5
2. Модели	7
2.1. Train-test и кросс-валидация	8
2.2. Random Forest Classifier	8
2.3. Support Vector Machine	9
2.4. XGBoost	11
2.5. TabNet	12
3. Результаты и выводы	13
Список литературы	14

Введение

Коричневые карлики — это субзвездные объекты. Их массы недостаточно для запуска и поддержания стабильного водородного синтеза, что приводит к их охлаждению с течением времени. В спектральной классификации коричневые карлики имеют спектральные типы L, T и Y. Коричневые карлики имеют низкие температуры, поэтому они довольно слабые в видимом диапазоне спектра, а пик интенсивности излучения приходится на инфракрасный диапазон.

Согласно исследованиям ([Mužić et al.(2017)]) количество коричневых карликов в Галактике составляет от 25 до 100 миллиардов объектов (при общем числе объектов от 100 до 500 миллиардов). Однородные и полные выборки коричневых карликов необходимы для разного рода исследований: кинематические исследования, исследования двойных звезд с коричневыми карликами, исследование параметров Галактики. Для моделирования наблюдаемых величин также необходимо уточнить зависимости спектральный тип – светимость и спектральный тип – плотность объектов в ближайшей окрестности.

Существует еще одно загадочное явление, связанное с коричневыми карликами, называемое L/T-переходом. По мере того, как мы движемся к более низким температурам на диаграмме Герцшпрунга-Рассела, цвет коричневых карликов становится значительно более синим, нарушая монотонное изменение цвета. При этом они становятся ярче, также нарушая монотонную тенденцию изменения блеска с уменьшением температуры. Это явление связано, по всей видимости, с изменением свойств атмосферы коричневых карликов. Некоторые атмосферные модели предполагают, что это происходит из-за опускания облаков пыли через фотосферу, что приводит к быстрому и резкому изменению цветовых характеристик. Есть также модели, которые связывают все с химической нестабильностью в атмосфере. Еще одним предложением являются так называемые модели «пятнистых облаков». Они мотивированы наблюдаемой изменчивостью коричневых карликов L- и T-типов. Спектральные исследования коричневых карликов в силу их слабости довольно трудоемки. По этой причине сложно создать большую надежную выборку коричневых карликов спектроскопическими наблюдениями представляется пока недоступным.

Предпринимались многочисленные попытки поиска и создания набора коричневых карликов, используя их фотометрические параметры в качестве решающего правила. Например, [Skrzypczak et al.(2016)] успешно использовали данные трех обзоров: SDSS, UKIDSS и WISE. Для поиска коричневых карликов они использовали следующие ограничения: $(Y - J)_{Vega} > 0.8$, $J < 17.5$. В результате удалось обнаружить порядка 1300 коричневых карликов на площади 3000 квадратных градусов (около 7,5% небесной сферы). Еще один недавний успешный пример: [Carnero Rosell et al.(2019)]. Они также пользовались решающим правилом, но использовали обзоры DES, VHS и WISE. В этом случае использовались следующие

правила: $(i - z) > 1.2$, $(z - Y) > 0.15$, $(Y_{AB} - J_{Vega}) > 1.6$, $z < 22$. Авторы утверждают, что таким образом удалось найти почти 12 тысяч коричневых карликов на площади 2400 квадратных градусов (около 5,8% небесной сферы).

Методы машинного обучения для классификации астрономических объектов все чаще используются в исследованиях. Например, [Maravelias et al.(2022)], скомбинировали метод опорных векторов, случайный лес и многослойный перцептрон для классификации массивных звезд в ближайших галактиках. Точность применения к тестовому датасету составила 83%. Применение к другим галактикам (не входящим в датасет) IC 1613, WLM и Sextans A показало результат точности на уровне 70%, что авторы связывают с отличающейся от тренировочного набора металличностью и эффектами поглощения. Пропущенные данные были заполнены простыми средними значениями, в качестве альтернативы использовался IterativeImputer, который показал лучшую эффективность. Еще одна работа, посвященная классификации астрономических объектов методами машинного обучения - [Lu et al.(2021)] - классифицирует звезды по изображению в трехцветной фотометрии. В данной работе использовалась CNN для того, чтобы извлечь признаки из изображения, а затем использовался метод опорных векторов непосредственно для классификации. Заявленная точность достигает 79%, работа с пропущенными данными не проводилась.

В данной работе мы используем методы машинного обучения (Random Forest, Support Vector Machine, XGBoost, TabNet) для выделения коричневых карликов среди других объектов и сравниваем его с классическими решающими правилами [Burningham et al.(2013)] и ([Carnero Rosell et al.(2019)]). Целью работы является создание метода для поиска коричневых карликов в больших фотометрических обзорах. То есть по набору блесков и цветов объекта метод должен определять, является ли данный объект коричневым карликом. Это поможет облегчить поиск и исследование коричневых карликов.

1. Данные

Первоначально в работе планировалось провести кросс-сопоставление списка L, T и Y коричневых карликов [Kirkpatrick et al.(2021)] с данными нескольких оптических и инфракрасных обзоров. Но в виду трудности кросс-сопоставления быстро движущихся объектов, было принято решение начать работу над обучением классификатора на уже размеченных данных.

Основу датасета составляют L и T коричневые карлики из каталога [Best et al.(2018)]. Каталог содержит информацию о 1601 коричневом карлике L и T типа и о 8287 красных карликах M типа, наиболее близком по физическим характеристикам к коричневым карликам спектральном классе, субкарликах и др. Каталог состоит

из фотометрических данных объектов - блесков (характеристика яркости объекта) в 12 фотометрических полосах и их погрешностей. Приведены пять значений блесков в оптическом спектре, полученных на телескопе Pan-STARRS 1 ([Chambers et al.(2016)]) в ходе миссии Pan-STARRS. Еще семь значений блеска - в инфракрасном спектре - взяты данных космической миссии 2MASS ([Cutri et al.(2003)]) и космической миссии WISE ([Cutri et al.(2021)]). Каталог также содержит астрометрическую информацию, положение объекта на небе, параллакс (расстояние до объекта) и собственное движение - скорость перемещения объекта на небесной сфере. Кроме того, он содержит ссылки на литературу, из которой взяты данные о собственных движениях и параллаксах.

Объекты L и T типов мы объявили объектами положительного класса. Для объектов отрицательного класса было выбрано около 650 красных карликов из того же каталога [Best et al.(2018)], а также около 1700 объектов других спектральных классов и классов светимости. Объекты других классов были выбраны из базы данных астрономических объектов Simbad (<http://simbad.cds.unistra.fr/simbad/>). Для них зачастую хорошо исследована спектральная классификация. Затем было проведено кросс-сопоставление выбранных из Simbad объектов с данными каталогов Pan-STARRS DR1, 2MASS и ALLWISE. Для каждого объекта внутри некоторого радиуса, называемого радиусом кросс-матчинга, вокруг его положения, данного в Simbad, искался парный объект в каждом из интересующих нас обзоре. Поскольку объекты в основном далекие и имеют небольшие собственные движения, радиус матчинга мы выбрали равным $1''$, что является разумным значением для большинства обзоров, в том числе, используемых в данной работе.

В результате получился список из 3983 объектов, 1601 из которых - целевого положительного класса. Данные имеют вид таблицы из 3983 строк и 14 колонок: в одной указана спектральная классификация объекта из литературы, 12 занимают фотометрические блески объектов в разных фотометрических полосах и в последней колонке - метка, принадлежит ли объект к целевому классу.

Данные содержат значительное количество пропущенных значений для объектов целевого класса в оптическом спектре (полосы g, r, i). Так как пик интенсивности и без того тусклых объектов приходится на инфракрасную часть спектра, блески в этих фотометрических полосах, скорее всего, находятся за пределом чувствительности телескопа. В полосах g и r данных Pan-STARRS значения пропущены практически для всех объектов, поэтому от этих данных пришлось совсем отказаться. В полосе i значения отсутствуют примерно для одной трети объектов положительного класса и незначительного количества объектов отрицательного класса. Значения блесков в этой полосе нам важны, в том числе для сравнения с классическими решающими правилами, поэтому их мы оставляем. В результате получается таблица из 3983 строк, соответствующих различным объектам, и 12 колонок, из которых 10 - признаки каждого объекта.

1.1. Генерация дополнительных признаков

В астрономии важную роль играет не только характеристика яркости объекта - его блеск, но и характеристика распределения энергии в спектре объекта - показатель цвета, который является разностью двух блесков. Для того, чтобы учесть это при классификации, мы добавили несколько колонок - классических показателей цвета, часто используемых в данной области: $(i - z)_{PS1}$, $(z - Y)_{PS1}$, $z_{PS1} - J_{2MASS}$, $Y_{PS1} - J_{2MASS}$, $J_{2MASS} - H_{2MASS}$, $H_{2MASS} - Ks_{2MASS}$, $W1 - W2$. Они так же часто используются в решающих правилах для отбора коричневых карликов среди остальных объектов.

Таким образом, после этой процедуры таблица содержит по 17 признаков для каждого из 3983 объектов. На Рис.1 показано, как выглядят объекты целевого класса по сравнению с объектами всех других классов в двумерном срезе пространства признаков.

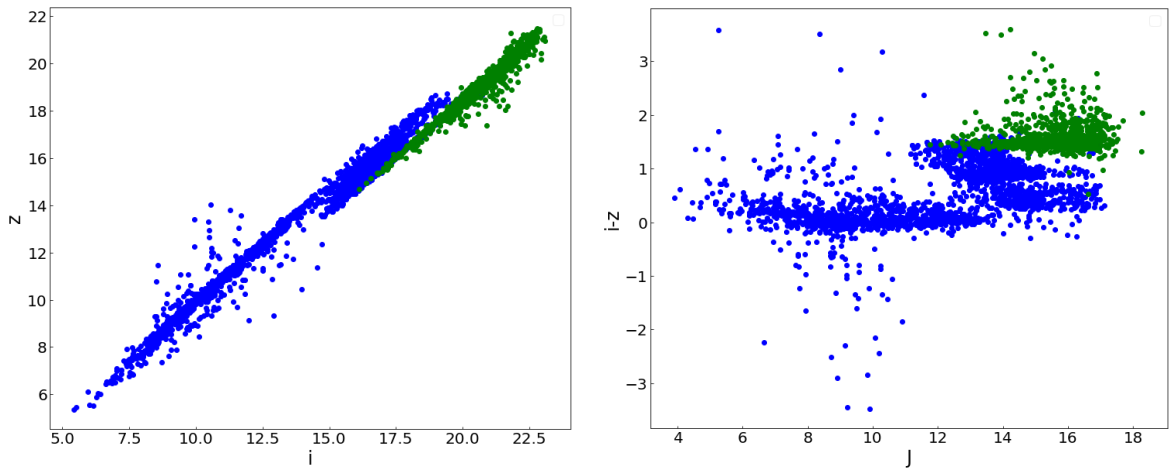


Рис. 1: Объекты разных классов на диаграммах блеск-блеск и блеск-показатель цвета. Зеленые точки - объекты положительного класса, синие - отрицательного.

1.2. Работа с пропущенными значениями

Как было сказано ранее, таблица имеет значительное количество пропущенных значений. В более длинноволновой части спектра это, скорее всего, связано с пределом чувствительности регистрирующего прибора: коричневые карлики являются довольно тусклыми объектами и максимум их излучения приходится на инфракрасную часть спектра. Пропущенные значения в более коротковолновой части спектра по-видимому связаны с некачественными измерениями или артефактами.

[Maravelias et al.(2022)] для заполнения пропущенных значений использовали метод заполнения средними и IterativeImputer библиотеки sklearn.impute. В данной

работе мы сразу отказались от заполнения средними значениями, так как метод является физически не обоснованным. В качестве метода заполнения пропущенных значений был выбран так же IterativeImputer. В методе IterativeImputer для заполнения отсутствующих значений применяется моделирование каждого признака с отсутствующими значениями в зависимости от других признаков циклическим способом. Этот метод сравнивался с методом KNNImputer из той же библиотеки sklearn.impute. IterativeImputer при схожей эффективности заполнения является все-таки более физически обоснованным в случае, когда пропущенные значения находятся на пределе регистрации. В случае KNNImputer метод усредняет значения признаков, ближайших к данному объекту в пространстве признаков, то есть он не может выйти за границы представленных в датасете значений признаков. В данной ситуации очевидно, что многие пропущенные значения в оптической части спектра как раз находятся за пределами представленных в датасете значений признаков.

Приведем пример работы метода на примере признака с наибольшим количеством пропущенных значений - i . Для проверки метода мы дополнительно выкинули значения блеска i для 170 объектов. На Рис.2 (слева) представлена диаграмма $Y - i$. Синие точки - объекты, для которых блеск i присутствует в датасете. Зеленые точки - объекты, для которых это значение в датасете отсутствовало изначально. Красные точки - объекты, для которых блеск i мы выкинули вручную. Блеск Y наиболее представлен в датасете, поэтому его значения не искажают картину. Как видно, красные и зеленые точки хорошо ложатся на общий тренд за исключением области, в которой синих точек не хватает ($Y > 20$). С помощью подбора параметров метода мы получили результат, наименее отклоняющийся от общего тренда.

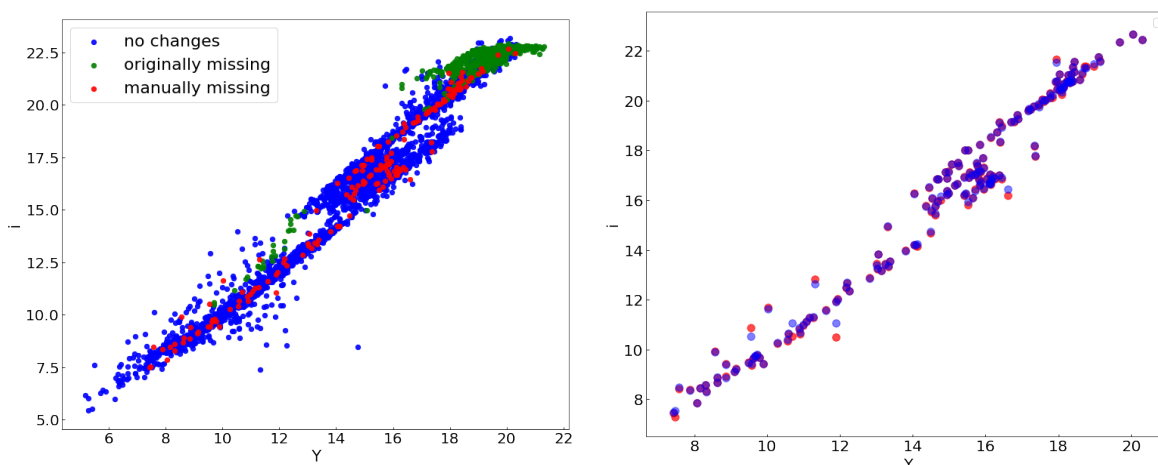


Рис. 2: Сравнение оригинальных данных (голубые точки) и вставленных значений (красные точки) методом IterativeImputer

На Рис.2 (справа) представлено сравнение значений блеска i "реального" и заполненного методом IterativeImputer. Синие точки соответствуют реальным значе-

ниям блеска i , красные - заполненным. По блеску Y эта выборка является полной. Для единичных объектов разница может достигать половины звездной величины, в остальных случаях значения предсказываются, можно сказать, идеально.

Несмотря на то, что вычисленные в разделе 1.1 показатели цвета непосредственно связаны со значениями блесков, было решено применять к ним IterativeImputer независимо. Это позволяет добиться лучших результатов и избежать больших ошибок в вычислении показателей цвета. В случае, если бы мы сначала применяли заполнение к блескам, а затем вычисляли бы показатели цвета, отклонения при заполнении блесков могли бы привести к ошибкам в показателях цвета, примерно в два раза превышающим ошибку заполнения каждого из блесков. Независимое заполнение значениями позволяет получить меньшие ошибки.

Кроме того, исследовалась возможность аугментации данных гауссовым шумом для того, чтобы увеличить размер датасета. К сожалению, данные после аугментации не удалось подружить с IterativeImputer, предсказания для искусственно пропущенных значений стали хуже, и так же стали немного хуже результативности моделей, поэтому от такой опции на данный момент пришлось отказаться.

2. Модели

В ходе работы было опробовано четыре подхода: Random Forest Classifier, Support Vector Machines, XGBoost и TabNet. Мы сравниваем их с классическими решающими правилами [Carnero Rosell et al.(2019)] и [Burningham et al.(2013)], в качестве метрики был выбран коэффициент корреляции Метьюса, поскольку он учитывает и ложноположительные, и ложноотрицательные предсказания. Решающие правила и результат их применения к тестовому датасету суммированы в Таб.1.

Автор	Правило	MCC
Carnero Rosell et al. (2019)	$(i - z) > 1.2, (z - Y) > 0.15,$ $(Y_{AB} - J_{Vega}) > 1.6, z < 22$	0.84
Burningham et al. (2013)	$(z - J)_{Vega} > 2.5, J < 17.5$	0.87

Таблица 1: Решающие правила из литературы

На Рис. 3 представлены матрицы ошибок, отражающие количество верно классифицированных, ложноположительных и ложноотрицательных объектов для обоих правил. Несмотря на то, что результативность решающих правил достаточно высока, количество ложноположительных и ложноотрицательных классификаций растет с ростом числа объектов и это становится важным, когда объектов у нас миллионы и больше, как в большинстве современных небесных обзоров (PanSTARRS - 1.9 млрд объектов, 2MASS - 470 млн объектов, WISE - 560 млн объектов), поэтому есть смысл бороться за проценты.

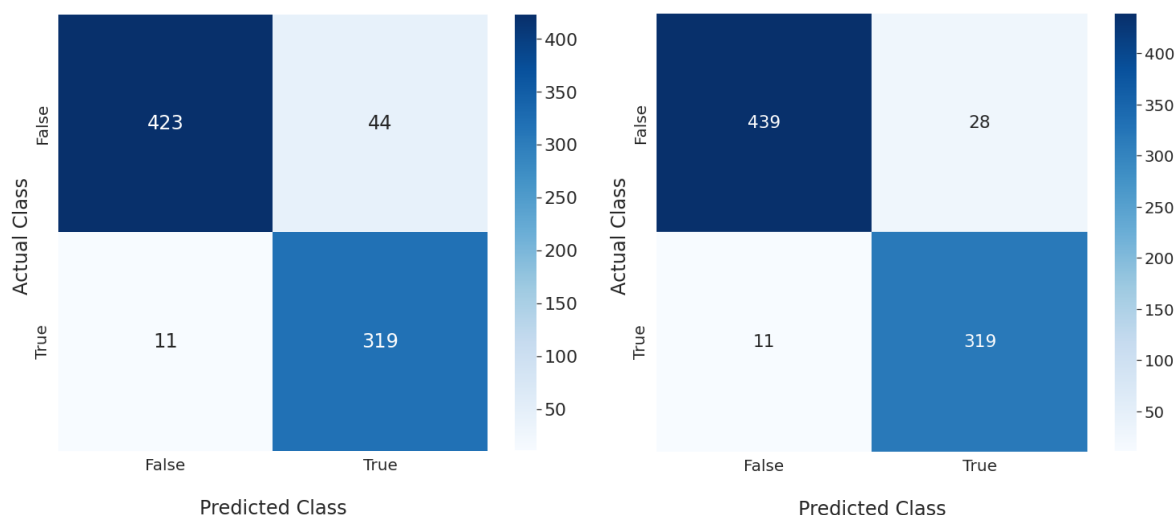


Рис. 3: Матрицы ошибок решающих правил на валидационном датасете. Carnero Rosell et al. (2019) справа и Burningham et al. (2013) слева.

2.1. Train-test и кросс-валидация

Данные разбивались на обучающий, валидационный и тестовый набор в отношении 6:2:2. Гиперпараметры моделей подбирались на валидационном наборе с помощью *optuna*. А итоговая результативность модели проверялась на тестовом наборе. Так как для каждого из объектов представлен набор данных с одних и тех же трех телескопов, то разбиение оправдано с научной точки зрения. К данным также применялось скалирование с помощью функции *StandardScaler*.

2.2. Random Forest Classifier

Метод решающего дерева довольно сходен по своей концепции с классическими решающими правилами, которые традиционно используются в астрономии для классификации объектов. И хотя автоматизированные деревья решений могут быть намного более эффективными, чем классические решающие правила, они склонны к переобучению, т. е. слишком хорошо изучают данные, на которых обучаются, и могут дать сбой при применении к данным, которые они раньше не видели. Решением такой проблемы может служить случайный лес (RF) - ансамбль из решающих деревьев. В таком случае решение о том, к какому классу принадлежит объект принимается на основании того, за какой класс проголосовало большее число деревьев.

С помощью *optuna* подбирались максимальная глубина дерева, минимальное количество листьев в дереве и максимальное количество признаков в узле. После оптимизации гиперпараметров случайный лес дает результат до $MCC = 0.974$ на тестовом датасете. Подобранные гиперпараметры:

```
max_depth = 15
min_samples_leaf = 10
```

`max_features = 2`

На Рис.4 на примере двумерного среза показана разделяющая граница между классами, определенная моделью RF.

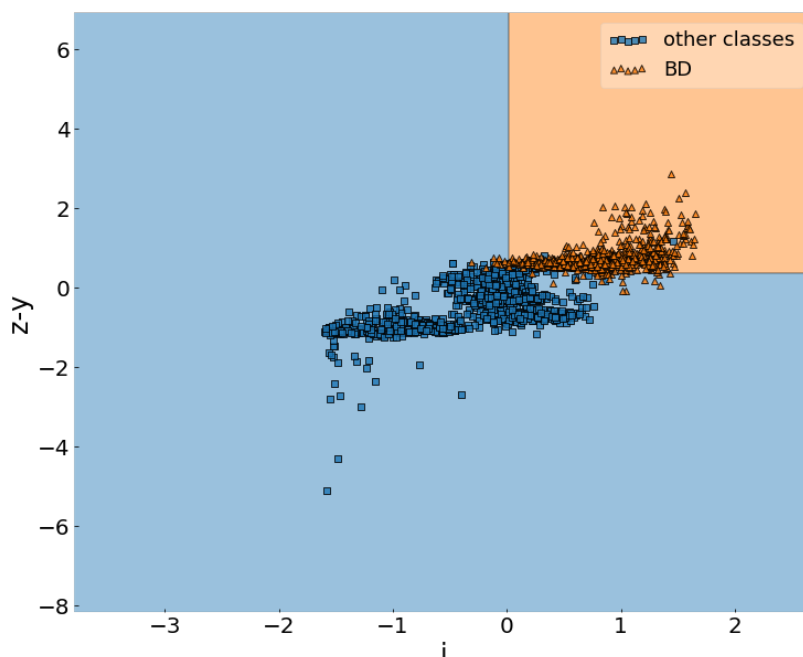


Рис. 4: Срез разделяющей границы в пространстве признаков по модели RF.

Кроме этого мы также посмотрели на важность признаков, которые модель использует при классификации. Диаграмма важности признаков для RF представлена на Рис.5 (слева). По диаграмме видно, что наиболее важным является признак *PS1imag* (он же просто *i*). Это логично, но не слишком хорошо, поскольку величина блеска зависит в том числе от расстояния до объекта и может быть разной для абсолютно одинаковых объектов, если один ближе, а другой дальше. Нам бы не хотелось полагаться на него в первую очередь. К тому же, этот признак наименее представлен в оригинальном датасете и, соответственно, имеет больше всего вставленных значений. Если убрать из обучения признаки *PS1imag*, *PS1zmag*, *PS1ymag*, то модель начинает опираться в основном на цвета $z - Y$ и $i - Y$, что хорошо, так цвета практически не зависят от расстояний и являются хорошим маркером, диаграмма важности признаков в таком случае представлена на Рис.5 (справа). При этом результативность немного снижается до $MCC = 0.966$. Поскольку существует такая проблема, все дальнейшие модели мы будем рассматривать для двух случаев: со всеми признаками и без *PS1imag*, *PS1zmag*, *PS1ymag* и, соответственно, сравнивать модели в одинаковых условиях.

2.3. Support Vector Machine

Метод опорных векторов - это широко применяемый и хорошо разработанный метод. Принцип метода опорных векторов заключается в нахождении линии, по-

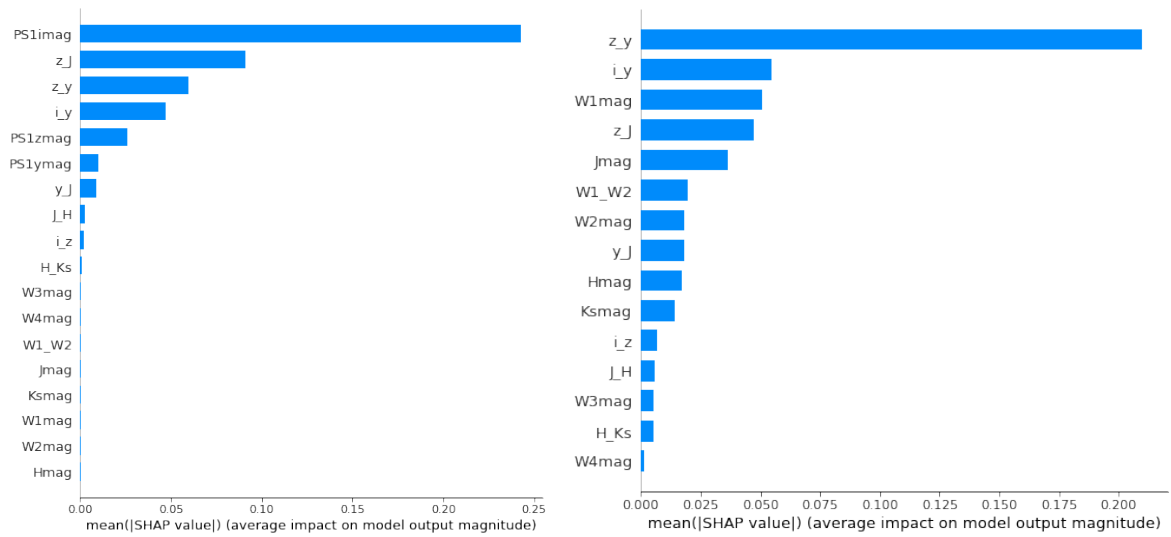


Рис. 5: Важность признаков для модели RF на всех признаках (справа) и без части признаков (слева)

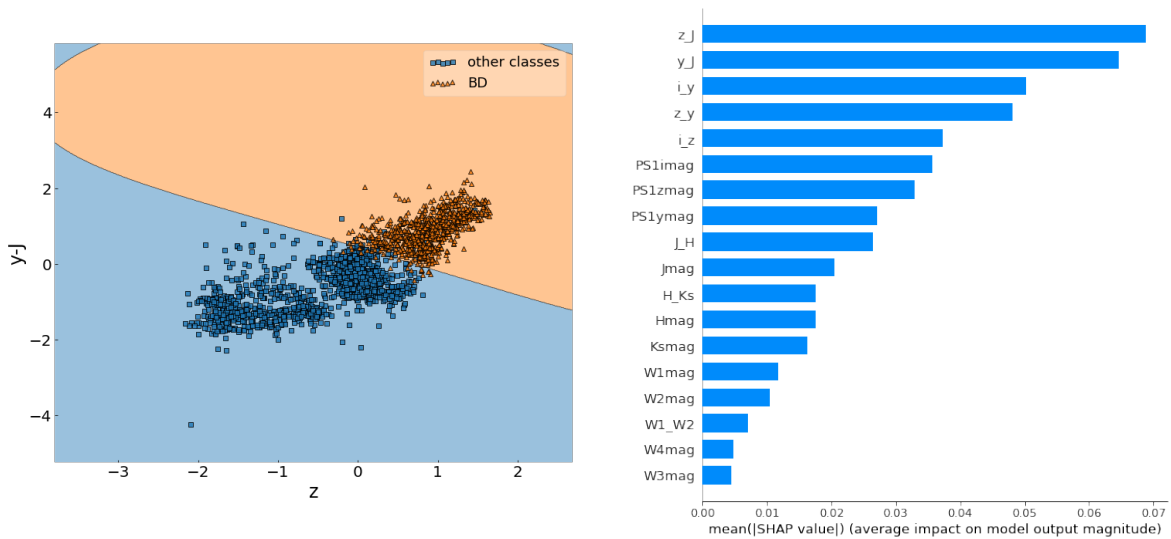


Рис. 6: Срез разделяющей границы в пространстве признаков по модели SVM и важность признаков.

верхности или гиперповерхности, которая бы разделяла классы в пространстве признаков. В процессе подбора максимизируется расстояние от каждой точки до границы раздела (опорный вектор).

Подбор гиперпараметров с помощью optuna дал следующие параметры:

```
kernel = 1.1785578339058878
C = rbf
decision_function_shape = ovr
coef0 = 4.36590388210411
gamma = scale
```

На тестовом датасете метод опорных векторов дает до $MCC = 0.965$, что почти идентично случайному лесу. Если обучать модель на 14 признаках (без оптических

блесков), то результат получается почти идентичный.

На Рис.6 слева показана разделяющая граница, построенная моделью SVM, на примере двумерного среза. Срез отличается от приведенного на аналогичном рисунке для RF, так как разные формы разделяющих поверхностей хорошо показывают себя на разных признаках. Стоит отметить, что это только срез, в котором остальные значения признаков взяты в некоторой окрестности среднего, поэтому большое количество точек, которые попали не в ту область, не означает фактическую мисклассификацию.

Справа на Рис.6 показана диаграмма важности признаков, здесь хорошо то, что весь топ важных признаков занимают цвета, а не блески и это именно то, что мы хотим от модели.

2.4. XGBoost

Бустинг - очень популярный алгоритм машинного обучения. Он представляет собой тип ансамблевого обучения, который использует результат предыдущей модели в качестве входных данных для следующей. Вместо того, чтобы обучать модели по отдельности, бустинг обучает модели последовательно, причем каждая новая модель обучается для исправления ошибок предыдущих. На каждой итерации результатам, предсказанным правильно, присваивается меньший вес, а результатам, предсказанным неправильно - больший вес. Затем он использует средневзвешенное значение для получения окончательного результата.

Также бустинг нам интересен с той точки зрения, что зачастую в алгоритм встроен некоторый принцип обработки пропущенных значений, что в нашей работе очень важно. Две популярных модели бустинга - это CatBoost и XGBoost. В CatBoost встроено только заполнение пропущенных значений некоторым конкретным числом, а вот XGBoost использует более хитрую стратегию: каждому узлу приписывается решение по умолчанию и это во многих случаях хорошо работает. Поэтому для работы мы выбрали XGBoost и также сравнили результативность модели на заполненных методом по умолчанию пропущенных значениях и на заполненных IterativeImputer.

С помощью optuna подобрали гиперпараметры:

```
max_depth = 24
learning_rate = 0.7643065952721012
subsample = 0.5361396679408698
gamma = 0.6069151786267634
```

На тестовом датасете с дефолтным алгоритмом работы с пропущенными значениями XGBoost дает $MCC = 0.96$. При обучении и тесте на данных, пропущенные значения в которых мы заполнили методом IterativeImputer самостоятельно, ре-

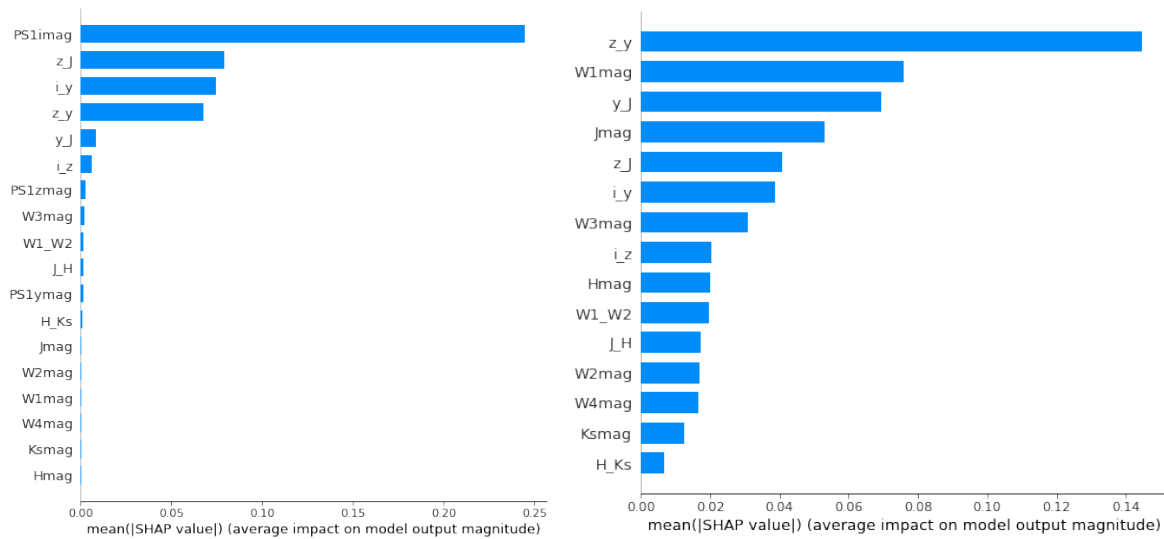


Рис. 7: Важность признаков для модели XGBoost на всех признаках (справа) и без части признаков (слева)

зультативность доходит до $MCC = 0.98$. Таким образом, можно сделать вывод, что IterativeImputer в данном случае не только является более обоснованным методом, но и лучше влияет на производительность модели.

Так же, как и в случае Random Forest, XGBoost при обучении на всех признаках, большую важность придает блескам в оптической части спектра: $PS1imag$, $PS1zmag$, $PS1ymag$ (см. Рис.7). Если убрать только один из них, модель начинает ориентироваться на другой, поэтому нужно убирать все три. В этом случае главным становится признак $z - Y$, при этом результативность даже возрастает до $MCC = 0.987$

2.5. TabNet

TabNet ([Arik & Pfister(2019)]) - это нейронная сеть глубокого обучения, использующая внимание для отбора на каждом этапе принятия решения, что используются только наиболее полезные признаки. При этом выбор признаков зависит от объекта, и например, он может быть разным для каждой строки обучающего набора данных. В конце можно посмотреть, на какие признаки модель больше всего ориентировалась. Внешний отбор признаков, таким образом, не производился. TabNet состоит из нескольких шагов, каждый шаг представляет собой блок компонентов, при этом количество шагов является гиперпараметром. Каждый шаг получает свой собственный голос в финальной классификации, что имитирует классификацию с помощью ансамбля.

Гиперпараметры модели были подобраны на optuna:

```
n_da = 32
n_steps = 3
n_shared = 3
```

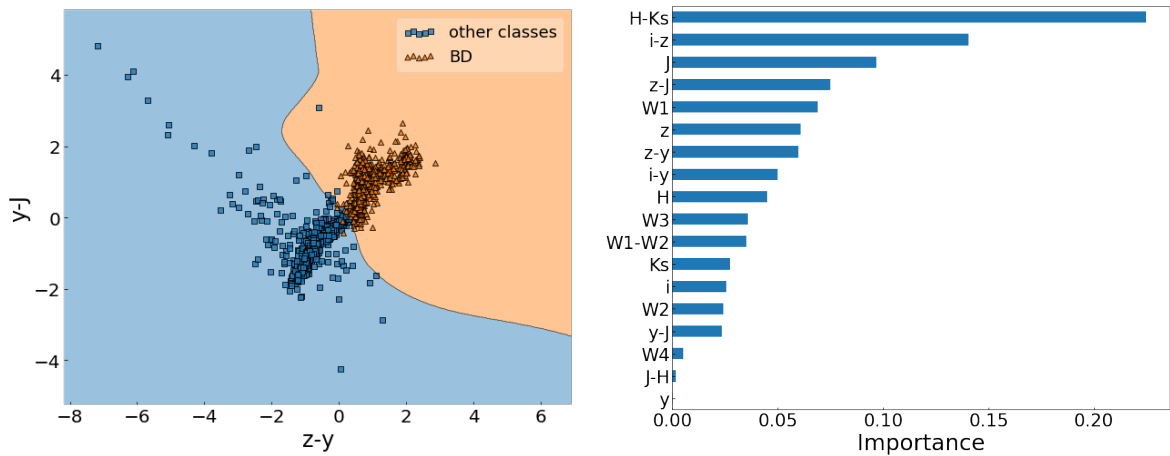


Рис. 8: Срез разделяющей границы в пространстве признаков по модели TabNet и важность признаков.

$\text{gamma} = 1$

затем модель была обучена с использованием оптимизации на основе градиентного спуска, в качестве оптимизатора использовался Adam. Кроме того, модель обучается, проверяя результаты на валидации.

На тестовом датасете TabNet показывает эффективность $MCC = 0.966$, что также в целом совпадает с результатами RF, XGBoost и SVM. В случае обучения только на части признаков TabNet показывает результат чуть хуже, до $MCC = 0.96$. Итоговая важность признаков представлена на Рис.8 (справа). На Рис.8 слева показана разделяющая поверхность в проекции на срезе самых важных по версии модели признаков. Стоит отметить, что второй и четвертый по важности признаки, также отражены и в решающих правилах Таб.1, поэтому этот результат является ожидаемым. А вот первый по важности признак довольно неожиданный, в литературе нигде не встречается разделение именно по нему.

3. Результаты и выводы

В данной работе мы составили датасет из коричневых карликов L и T типа (отмеченных, как положительный класс) и объектов других спектральных классов (отмеченных, как отрицательный класс) на основе литературных источников. Мы обучили на этих данных четыре модели: Random Forest Classifier, SVM Classifier, XGBoost и TabNet Classifier. Все четыре модели на полном наборе данных дали практически одинаковый результат, но некоторые показали себя чуть лучше. Random Forest и XGBoost в своих решениях очень сильно опираются на значения блесков. Такая ситуация обоснована, так как карлики являются более слабыми объектами и в среднем более тусклые, чем звезды. Тем не менее, очень далекие объекты тоже могут выглядеть слабыми и такие объекты могут быть ошибочно

классифицированы, как коричневые карлики на основании этого признака. Если убрать из обучающего набора блески в оптической области спектра, XGBoost показывает даже лучшие результаты, чем на полном наборе признаков.

SVM Classifier идеально показывает себя с точки зрения признаков, которые считает важными, но обладает меньшей, по сравнению с другими моделями результативностью.

Доверительные интервалы для качества полученных моделей представлены на Рис 9 (качество моделей на всей признаках) и Рис 10 (качество на части признаков). Как видно, SVM и TabNet даже на всех признаках не превосходят XGBoost в случае обучения на чати признаков. Из этого можно сделать вывод, что XGBoost является лучшим решением для данной задачи.

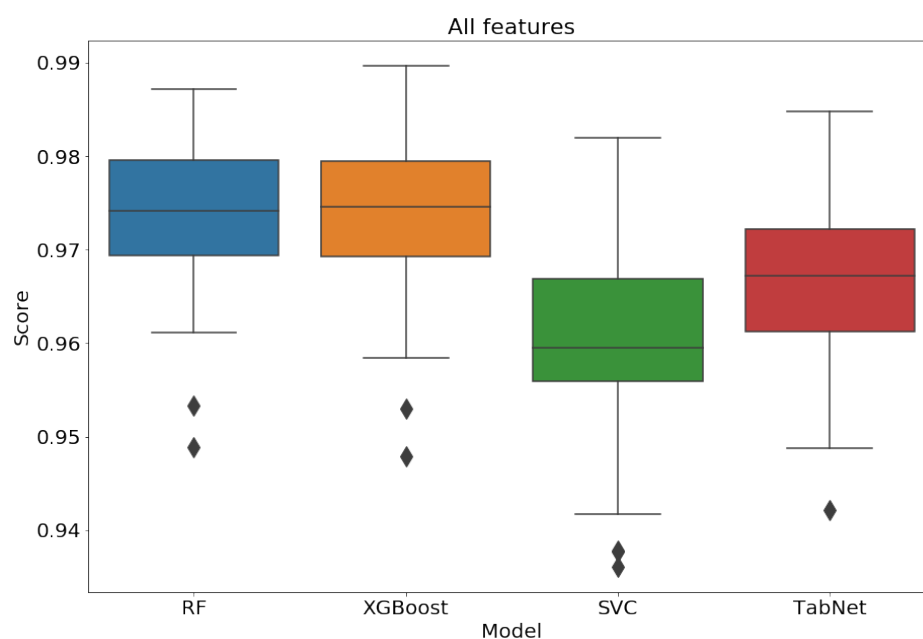


Рис. 9: Доверительные интервалы для моделей в случае обучения на всех признаках.

Код выложен по ссылке: https://github.com/iamaleksandra/Brown-Dwarfs-NN/blob/main/bd_open.ipynb. Статью по результатам исследования планируется опубликовать в журнале Astronomy&Computing, поскольку он идеально подходит по тематике.

Список литературы

- [Arik & Pfister(2019)] Arik, S. O., & Pfister, T. 2019, arXiv e-prints, arXiv:1908.07442
- [Best et al.(2018)] Best, W. M. J., Magnier, E. A., Liu, M. C., et al. 2018, , 234, 1
- [Burningham et al.(2013)] Burningham, B., Cardoso, C. V., Smith, L., et al. 2013, , 433, 457

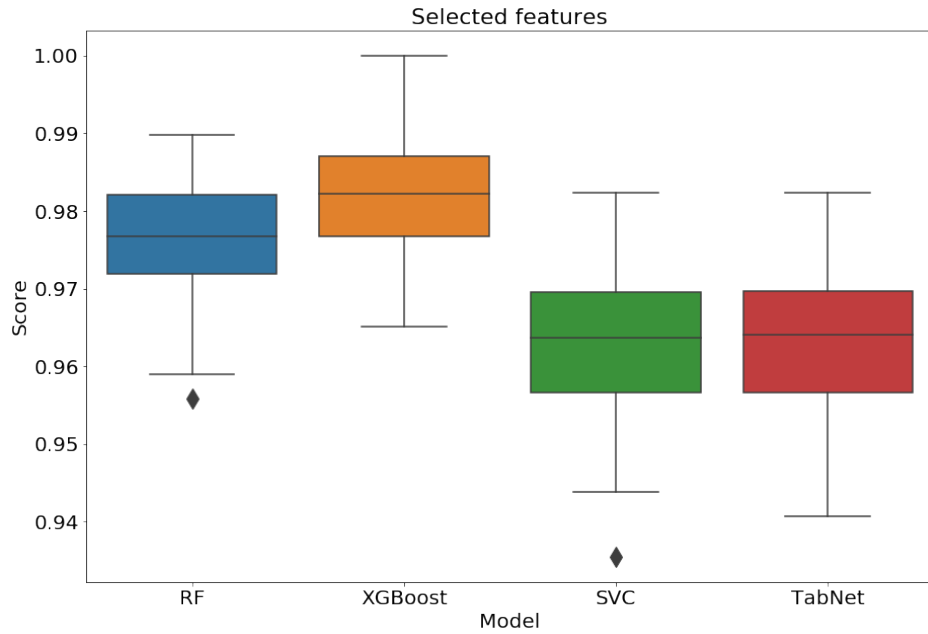


Рис. 10: Доверительные интервалы для моделей в случае обучения части признаков, без *PS1imag*, *PS1zmag*, *PS1ymag*.

[Carnero Rosell et al.(2019)] Carnero Rosell, A., Santiago, B., dal Ponte, M., et al. 2019, , 489, 5301

[Chambers et al.(2016)] Chambers, K. C., Magnier, E. A., Metcalfe, N., et al. 2016, arXiv e-prints, arXiv:1612.05560

[Cutri et al.(2003)] Cutri, R. M., Skrutskie, M. F., van Dyk, S., et al. 2003, VizieR Online Data Catalog, II/246

[Cutri et al.(2021)] Cutri, R. M., Wright, E. L., Conrow, T., et al. 2021, VizieR Online Data Catalog, II/328

[Kirkpatrick et al.(2021)] Kirkpatrick, J. D., Gelino, C. R., Faherty, J. K., et al. 2021, , 253, 7

[Lu et al.(2021)] Lu, Y.-K., Qiu, B., Luo, A. L., et al. 2021, , 507, 4095

[Maravelias et al.(2022)] Maravelias, G., Bonanos, A. Z., Tramper, F., et al. 2022, arXiv e-prints, arXiv:2203.08125

[Mužić et al.(2017)] Mužić, K., Schödel, R., Scholz, A., et al. 2017, , 471, 3699

[Skrzypek et al.(2016)] Skrzypek, N., Warren, S. J., & Faherty, J. K. 2016, VizieR Online Data Catalog, J/A+A/589/A49