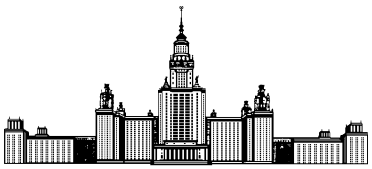
Московский государственный университет имени М. В. Ломоносова



Факультет Вычислительной Математики и Кибернетики

Кафедра Математических Методов Прогнозирования

# КУРСОВАЯ РАБОТА СТУДЕНТКИ 317 ГРУППЫ

## «Методы отбора признаков»

Выполнила:

студентка 3 курса 317 группы

*Рысьмятова Анастасия Александровна*

Научный руководитель:

д.ф-м.н., профессор

*Дьяконов Алксандр Геннадьевич*

Москва, 2015

# Содержание

# 1 Введение

На этапах постановки задачи машинного обучения и формирования данных не всегда понятно, какие признаки важны для построения оптимального алгоритма, поэтому часто в данных встречается много избыточной (шумовой) информации. Появление шумовых признаков ухудшает качество работы алгоритма и замедляет его работу. Поэтому в большинстве случаев перед решением задачи классификации, регресии или прогнозирования необходимо выбрать те признаки, которые наиболее информативны.

Правильный выбор признаков может быть более значимой задачей, чем уменьшение времени обработки данных, или улучшения точности классификации. К примеру, в медицине [1], нахождение минимального набора признаков, который является оптимальным для задачи классификации, может быть полезным для разработки диагностического теста.

Отбор важных признаков (например, отбор генов, соответствующих определенному типу рака) может помочь расшифровать механизмы, лежащие в основе проблемы, представляющей интерес для исследования.

В данной работе описаны основные методы отбора признаков, а также приведен пример работы некоторых из них. Показаны результаты работы некоторых из алгоритмов отбора признаков для задачи определения факторов, влияющих на стоимость арендной платы нежилых помещений в США.

# 2 Методы отбора признаков

Метод отбора может быть реализован путем полного перебора признаков, то есть, проверив все возможные наборы, выбрать те признаки, на которых ошибка минимальна. Такой метод прост в реализации, но он совершенно неэффективен на больших данных, поэтому в этом случае чаще всего используются другие алгоритмы. Существуют три основных класса алгоритмов отбора признаков - *фильтры, обертки* и *встроенные алгоритмы* [4].

## 2.1 Фильтры

*Фильтры (filters)* основаны на некоторых показателях, которые не зависит от метода классификации. Например, такие как корелляция признаков с целевым вектором, критерии информативности. Они применяются до классификации. Одним из приемуществ фильтрации является то, что она может быть использована в качестве предварительной обработки для уменьшения размерности пространства и преодоления переобучения. Методы фильтрации, как правило, быстро работают. Фильтры используются для отбора признаков в кластеризации, для построения начального приближения [2]. К сожалению такие методы не предназначены для обнаружения сложных связей между признаками, и, как правило, не являются достаточно чувствительными для выявления всех зависимостей в данных.

## 2.2 Встроенные алгоритмы

*Встроенные алгоритмы (embedded algorithms)* выполняют отбор признаков во время процедуры обучения классификатора, и именно они явно оптимизируют набор используемых признаков для достижения лучшей точности [10]. Преимущества встроенных алгоритмов в том, что как правило они находят решения быстрее, избегая переподготовки данных с нуля, при этом пропадает необходимость разделять данные на обучающую и тестовую подвыборку. Однако, данные алгоритмы не универсальны.

## 2.3 Методы обертки

*Методы обертки (wrappers)* опираются на информацию о важности признаков полученную от некоторых методов классификации или регрессии, и поэтому могут находить более глубокие закономерности в данных, чем фильтры. Обертки могут использовать любой классификатор, который определяет степень важности признаков. Подробнее несколько алгоритмов обертки будут рассмотрены в этой работе далее.

# 3 Алгоритмы отбора признаков с помощью Random Forest

Random Forest — алгоритм машинного обучения, предложенный Leo Breiman и Adele Cutler. Представляет собой ансамбль многочисленных, чувствительных к обучающей выборке алгоритмов (деревьев решений). Данные алгоритмы имеют маленькое смещение. Смещение (bias) метода обучения — это отклонение среднего ответа обученного алгоритма от ответа идеального алгоритма. Каждый из этих классификаторов строится на случайном подмножестве объектов и случайном подмножетсве признаков. Пусть обучающая выборка состоит из $N$ примеров, размерность пространства признаков равна $M$, и задан параметр $m$. Запишем пошагово алгоритм Random Forest.

Все деревья ансамбля строятся независимо друг от друга по следующей процедуре:

1. Сгенерируем случайную подвыборку с повторением размером $n$ из обучающей выборки.

2.Построим решающее дерево, классифицирующее примеры данной подвыборки, причём в ходе создания очередного узла дерева будем выбирать признак, на основе которого производится разбиение, не из всех $M$ признаков, а лишь из $m$ случайно выбранных.

3.Дерево строится до полного исчерпания подвыборки и не подвергается процедуре прунинга (англ. pruning — отсечение ветвей)

Классификация объектов проводится путём голосования: каждое дерево ансамбля относит классифицируемый объект к одному из классов, и побеждает класс, за который проголосовало наибольшее число деревьев.

Алгоритм Random Forest может быть использован в задаче оценки важности признаков. Для этого необходимо обучить алгоритм на выборке и во время построения модели для каждого элемента обучающей выборки посчитать out-of-bag-ошибку. Пусть $X_n^l$ - бутстрапированная выборка дерева $b_n$. Бутстрэппинг представляет собой выбор $l$ объектов из выборки с возвращением, в результате чего некоторые объекты выбираются несколько раз, а некоторые — ни разу. Помещение нескольких копий одного объекта в бутстрапированную выборку соответствует выставлению веса при данном

объекте — соответствующее ему слагаемое несколько раз войдет в функционал, и поэтому штраф за ошибку на нем будет больше. Пусть $L(y, z)$ — функция потерь, $y_i$ — ответ на $i$-м объекте обучающей выборки, тогда out-of-bag-ошибка вычисляется по следующей формуле:

$$\text{OOB} = \sum_{i=1}^{l} L\left(y_i, \frac{1}{\sum_{n=1}^{N}\left[x_i \notin X_n^l\right]} \sum_{n=1}^{N}\left[x_i \notin X_n^l\right] b_n(x_i)\right)$$

Затем для каждого объекта такая ошибка усредняется по всему случайному лесу. Чтобы оценить важность признака, его значения перемешиваются для всех объектов обучающей выборки и out-of-bag-ошибка считается снова. Важность признака оценивается путем усреднения по всем деревьям разности показателей out-of-bag-ошибок до и после перемешивания значений. При этом значения таких ошибок нормализуются на стандартное отклонение. Случайный лес имеет еще некоторые преимущества для использования его в качестве алгоритма отбора признаков: он имеет очень мало настраиваемых параметров, относительно быстро и эффективно работает, что позволяет находить информативность признаков без значительных вычислительных затрат.

## 3.1 Boruta

Эвристический алгоритм отбора значимых признаков, основанный на использовании Random Forest [5]. Суть алгоритма заключается в том, что на каждой итерации удаляются признаки , у которых Z-мера меньше максимальной Z-меры среди добавленных признаков. Чтобы получить Z-меру признака, необходимо посчитать важность признака, полученную с помощью встроенного алгоритма в Random Forest, и поделить ее на стандартное отклонение важности признака. Добавленные признаки получаются следующим образом: копируются признаки имеющиеся в выборке, а затем каждый новый признак заполняется путем перетасовки его значений. В целях получения статистически значимых результатов эта процедура повторяется несколько раз, переменные генерируются независимо на каждой итерации.

Запишем пошагово алгоритм Boruta :

1. Добавить в данные копии всех признаков. В дальнейшем копии будем называть скрытыми признаками.

2. Случайным образом перемешать каждый скрытый признак.

3. Запустить Random Forest и получить Z-меру всех признаков.

4. Найти максимальную Z-меру из всех Z-мер для скрытых признаков.

5. Удалить признаки, у которых Z-мера меньше чем найденная на предыдущем шаге.

6. Удалить все скрытые признаки.

7. Повторять все шаги до тех пор пока Z-мера всех признаков не станет больше чем максимальная Z-мера скрытых признаков.

### 3.2 ACE

ACE (Artificial Contrasts with Ensembles)[6] - еще один алгоритм, который может быть использован для отбора признаков. Главная идея алгоритма ACE схожа с идеей алгоритма Boruta - каждый признак заполняется случайным образом, путем перетасовки его значений. На полученной выборке запускается Random Forest. Однако, в нем, в отличие от Boruta, не удаляются найденные признаки с наименьшей важностью, которые позволяют повысить качество измерений важных признаков. Наиболее важные признакие найденные алгоритмом ACE, наоборот удаляют, что позволяет алгоритму находить более тонкие закономерности. Удаленные признаки ACE выдает в качестве ответа.

## 4 Пример использования алгоритмов отбора признаков

### 4.1 Gene expression data

В статье [1] был рассмотрен пример работы алгоритма Boruta для задачи выявления различия между двумя подтипами лейкоза. Эта задача решалась до этого другими методами [7], что позволило сравнить результаты после использования Boruta. Данные имеют информацию о 38 больных. По каждому из которых описано 3051 генов. Для оценки качества алгоритма было добавлено еще 1000 полу-синтетических признаков, которые сгенерированны путем перетасовки 1000 выбранных случайно

имеющихся генов. Хороший алгоритм не будет выбирать сгенерированные признаки как важные. Значимость гена оценивалась с помощью алгоритма Boruta на основе Random Forest. Число деревьев в Random Forest варьировалось от 500 до 100 000. Каждый запуск повторили 15 раз.

Введем следующие обозначения:

$Dud2002$ - 91 выделенных генов в 2002 году , решение описано в статье [9]

$Gol1999$ - 50 выделенных генов в 1999 году , решение описано в статье [7]

$Dram2008$ - 30 выделенных генов в 2008 году , решение описано в статье [8]

$Bor500$ - 82 выделенных генов с помощью Boruta на основе Random Forest с 500 деревьями

$Bor100k$ - 261 выделенных генов с помощью Boruta на основе Random Forest с 100000 деревьями На Рис. 1 приведен график зависимости количества выбранных генов, от числа деревьев. Черными точками выделен алгоритм Boruta, а остальными цветами показан результат работы алгоритмов $Dud2002$ , $Gol1999$ , $Dram2008$ , которым на вход подавались признаки выделенные с помощью Boruta с различным числом деревьев.

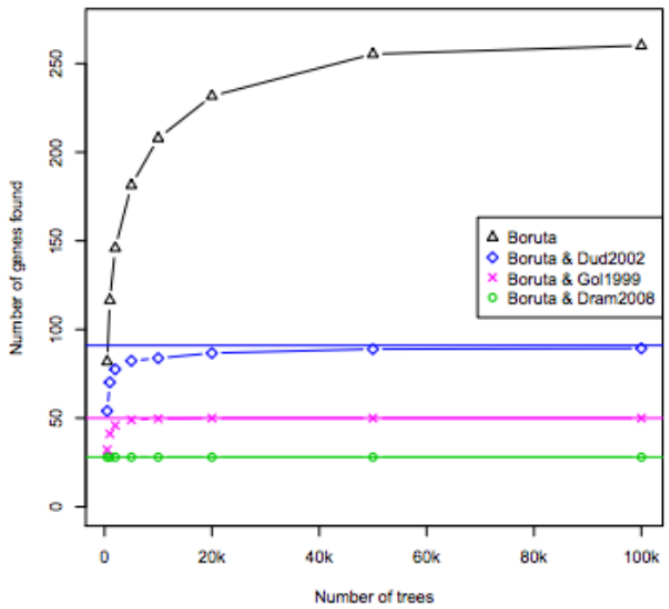



Рис. 1: зависимости количества выбранных генов, от числа деревьев

Видно, что с увеличением числа деревьев увеличивается количество выбранных генов. Сплошные горизонтальные прямые обозначают общее количество признаков выделенных данными алгоритмами. В результате получилось, что алгоритм ни разу не выбрал в качестве важных генов полу-синтетические данные, добавленные для проверки. На основе этого авторы данного эксперимента сделали вывод, что алгоритм Boruta эффективно справляется с задачей отбора признаков. Чтобы понять, как признаки, выделенные с помощью алгоритма Boruta, коррелируют с выделенными признаками ранее, и как множества выделенных ранее признаков коррелируют между собой, приведено графическое представление множества выделенных генов. На Рис. 2 изображено, как пересекаются выбранные множества важных генов у алгоритмов $Dud2002$, $Gol1999$, $Dram2008$, $Bor500$, $Bor100k$ . Площадь, покрытая голубыми, сиреневыми и зелеными полосами, представляют пересечения Bor100k с Dud2002, Gol1999 и Dram2008 наборов данных, соответственно.

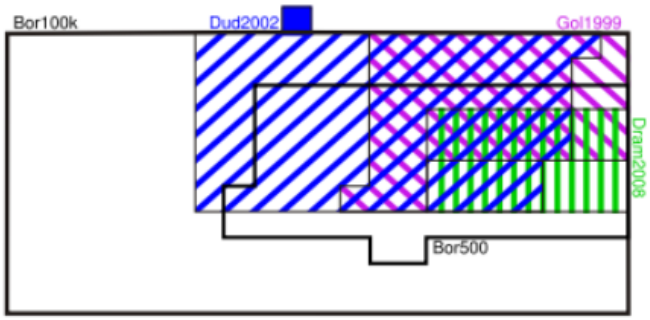



Рис. 2: пересечение важных генов у различных алгоритмов

Видно, что множество генов, выделенное с помощью Boruta с 500 деревьями, пересекается с Dud2002, Gol1999 и Dram2008. Гены, полученные в методах Dud2002, Gol1999 и Dram2008 слабо кореллируют между собой (площадь пересечения Dud2002, Gol1999 и Dram2008 небольшая). Гены выделенные с помощью Boruta с 100000 деревьями включают в себя все гены выделенные остальными алгоритмами. Данный эксперемент подтверждает, что все выделенные ранее признаки, действительно имеют значение. Более того, алгоритм выделил 150 новых генов. Результат данного эксперимента показывает, что чувствительность алгоритма Boruta зависит от количества деревьев используемых в Random Forest. Это происходит благодаря свойству

Z-меры. Она оценивается с помощью важности признаков, полученной из встроенного алгоритма в Random Forest, а важность пизнков это среднее снижение точности деревьев, которые используют данный признак. Поэтому актуальность гена можно узнать только при достаточно большом количестве деревьев.

# 5 Эксперименты с данными

## 5.1 Данные

Для изучения работы алгоритмов отбора признаков будут использованы данные о 2227 объектах недвижимости различного типа.

На основе предоставленных данных нужно предсказать стоимость арендной платы и определить признаки от которых наиболее сильно зависит арендная плата.
Данные имеют 19 признаков, из них 2 категориальных:
*SpaceType* – Тип здания в котором расположен объект, имеет 31 значение;
*LeaseType* – Несёт информацию о том какие расходы включены в арендную плату;

9 целочисленных :
*SpaceSize* – Размер помещения;
*Number of transport spots* – Количесво мест для стоянки транспорта;
*Population* – Количество населения в данном регионе;
*Landarea* – Площадь региона;
*Social chat score* – Престижность ( социальный балл);
*Average HH income* 2013 – Средний доход населения за 2013 год;
*Average salary of employees*($000s) – Средний даход рабочего;
*Average salary of employees in new businesses* – Средний даход рабочего в новом бизнесе;
*Number of new retail places* 2013−2010 – Число новых мест аренды за 2010-2013 год;

8 вещественных:
*Population change* 2013 – 2010 – Изменение числа населения в процентах;
*Density of people living in area* – Плотность людей проживающих в регионе;
*Density of people working in area* (*based on lat/lon*) – Плотность работающих людей;
*Total density* (*living* + *working*) – Сумма предыдущих двух признаков;
*Household size* – Размер складского помещения;
*Income change* 2013 – 2010 – Изменение дохода за 2010 - 2013 год;

*Change in % of bachelor degrees* – Изменение в процентах количества людей со степенью бакалавра;

*% of employees in new companies vs all* – Процент содрудников в новых компаниях;

## 5.2 Оценка важности признаков

### 5.2.1 Встроенный алгоритм

Изначальная выборка была разбита на три части, для использования кросс-валидации с тремя фолдами. Изначально категориальные признаки были закодированы набором булевых векторов, по одному на каждое значение признака. На Рис. 3 приведен график в котором для каждого из признаков вместе с набором булевых векторов показана их важность для трех частей выборки с помощью встроенного алгоритма Random Forest

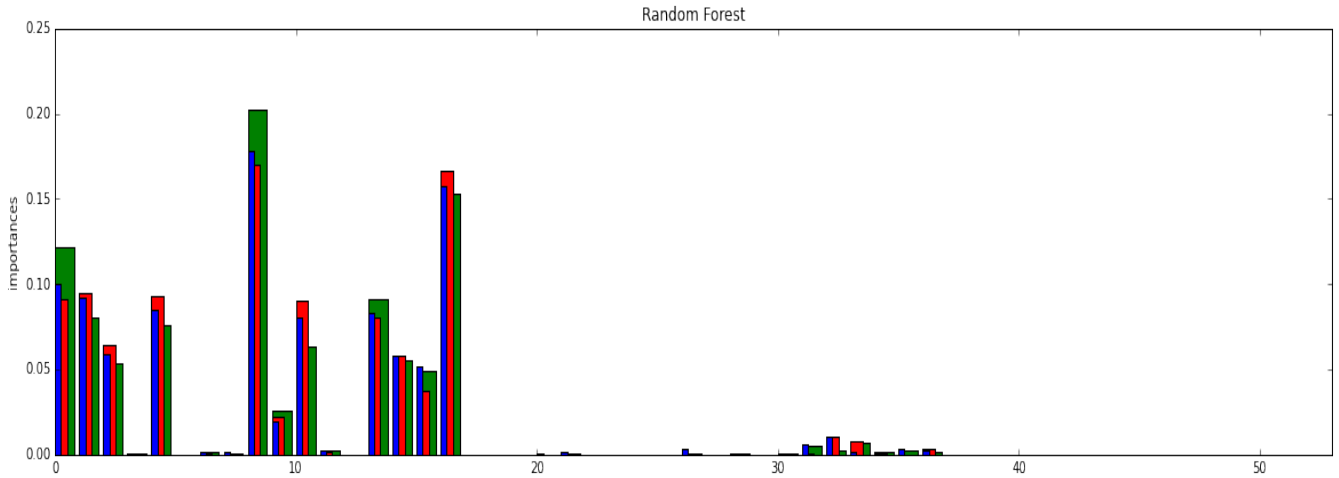



Рис. 3: важность признаков ( добавленный набор булевых векторов - это последние 37 признаков)

Важность добавленных бинарных признаков, почти у всех нулевая. А после многочисленных экспериментов, получилось, что набор булевых векторов для категориальных признаков, только ухудшал качество всех используемых алгоритмов, поэтому признаки соответствующие категориальным, были удалены из данных. На Рис.

4 приведен график в котором для каждого из 17 некатегориальных признаков показана их важность для трех частей выборки с помощью встроенного алгоритма в Random Forest

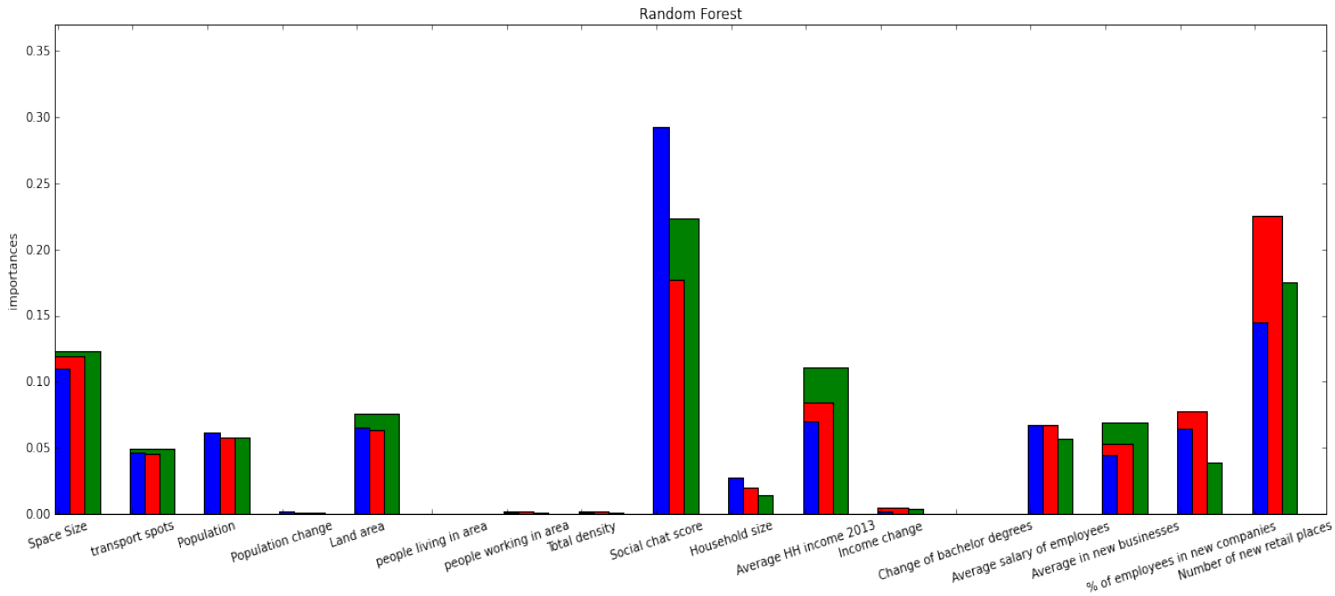



Рис. 4: важность признаков

На графике разными цветами указана важность признаков полученная по трем частям выборки. Максимальную важность в результате имеют следующии признаки:
*Social chat score* - Престижность ( социальный балл);
*Number of new retail places* 2013 – 2010 - Число новых мест аренды;
*SpaceSize* - Размер помещения;
*Average HH income* 2013 - Средний доход населения за 2013 год;
*Landarea* - Площадь региона;

Остальные признаки имели важность меньшую 0.05 (это значительно меньше важности выбранных признаков). Используя лишь те признаки, которые имели максимальную важность построен простой алгоритм, средняя ошибка которого по метрике MSE мало отличается от Random Forest. Данный алгоритм перебирает значение каждого важного признака у объекта и в зависимости от того, в каком интервале лежат признаки, присваивает целевой переменной определенную константу.

Параметры настраивались на двух третях случайно выбранных объектов выборки, а тестировались на оставшейся одной трети. После многочисленных экспериментов получилось, что Random Forest — это лучший алгоритм машинного обучения для решения задачи предсказания стоимости арендной платы, среди рассмотренных алгоритмов. На Рис. 4 приведены результаты тех алгоритмов, которые показали лучший результат.

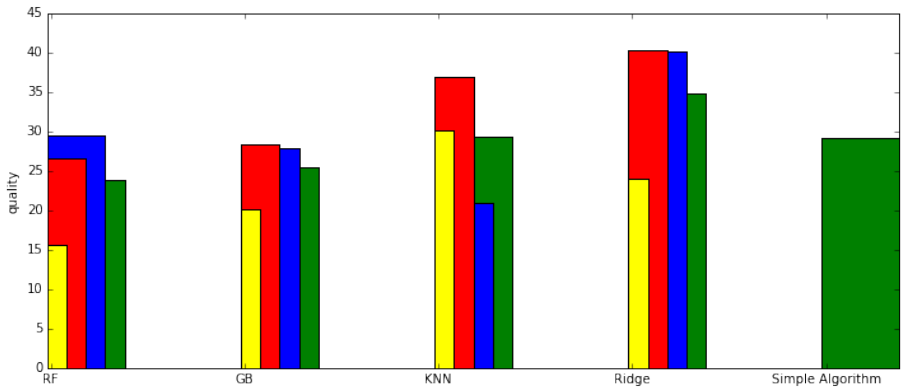



Рис. 5: результаты лучших алгоритмов

На графике указана ошибка алгоритмов по метрике MSE (Mean Squared Error) для каждого из трех фолдов и средняя ошибка по трем фолдам. Каждый алгоритм был применен сначала ко всей выборке, а затем только к признакам, которые были выбраны важными. В таблице приведена средняя ошибка алгоритмов по метрике MSE.

| | Качество на всех признаках | Качество на важных признаках |
|---|---|---|
| Random Forest | 24.1 | 23.8 |
| GB | 25.4 | 25.7 |
| KNN | 30.2 | 28.93 |
| Ridge | 35.0 | 34.7 |

Почти все используемые модели показали результат лучше на выделенных признаках, но для этого было необходимо снова настроить все параметры.

Реализация всех алгоритмов машинного обучения использовалась из библиотеки scikit-learn на языке python. Для каждого алгоритма были настроены все параметры с помощью функции GridSearchCV, но средняя ошибка по трем фолдам на настроенных параметрах почти не отличалась от ошибки на параметрах по умолчанию. Все

алгоритмы запускались и на выделенных признаках с помощью описанных алгоритмов отбора признаков, но средняя ошибка по трем фолмам оставалась примерно такой же. Так как простой алгоритм, основанный лишь на наиболее важных признаках, дал сравнительно неплохой результат, то можно сделать вывод, о том что встроенный в Random Forest алгоритм действительно хорошо справляется со своей задачей.

### 5.2.2 Boruta

К данным был применен алгоритм Boruta. На Рис. 5 показана важность признаков, которую определил алгоритм Boruta. Зеленым цветом обозначены значимые признаки. Важность признаков определялась на 200 итерациях алгоритма.

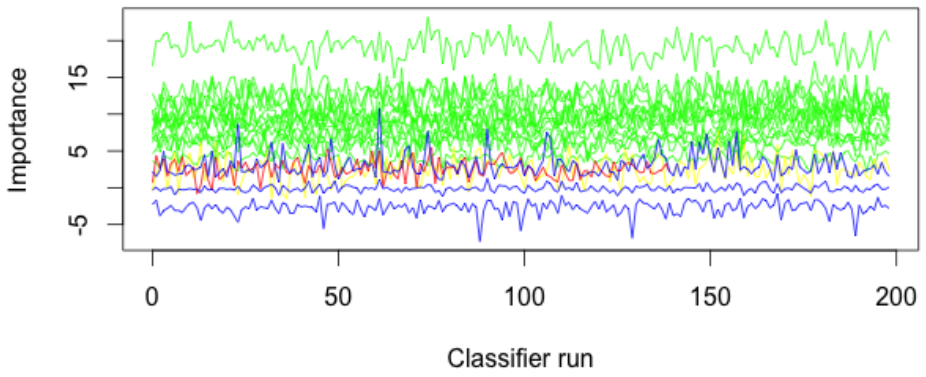



Рис. 6: важность признаков с помощью Boruta

Приведем наиболее значимые признаки, определенные с помощью Boruta.

| *Признак* | *Средняя значимость признака* |
|---|---|
| Space Size | 18.6 |
| Number of new retail places 2013-2010 | 12.8 |
| Social chat score | 12.3 |
| Density of people working in area based on lat lon. | 12.2 |
| Total density living working. | 11.3 |
| Average HH income 2013 | 9.4 |
| Household size | 8.7 |
| Population | 8.2 |

Получилось, что данный алгоритм выбрал почти все важные признаки, которые были выбраны с помощью встроенного алгоритма.

# 6 Заключение

Отбор признаков является важным этапом построения алгоритмов машинного обучения. Данный этап необходим, чтобы избавиться от шумовых признаков и благодаря этому улучшить качество и ускорить работу алгоритмов. Проведенные эксперименты подтверждают, что алгоритмы отбора признаков с помощью Random Forest эффективно справляется со своей задачей

# Список литературы


[1] The all relevant feature selection using random forest MB Kursa, WR Rudnicki arXiv preprint arXiv:1106.5112 (2011)

[2] Воронцов К. В.: Лекции по методам оценивания и выбора моделей (2007)

[3] Nilsson, R., Pena, J.M., Bjorkegren, J., Tegner, J.: Consistent feature selection for pattern recognition in polynomial time. The Journal of Machine Learning Research 8, 612 (2007)

[4] Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection. Journal of Machine Learning Research 3, 1157–1182 (2003)

[5] Kursa, M.B., Rudnicki, W.R.: Feature Selection with the Boruta Package. Journal Of Statistical Software 36(11) (2010)

[6] Tuv, E., Borisov, A., Runger, G., Torkkola, K.: Feature selection with ensembles, artificial variables, and redundancy elimination. The Journal of Machine Learning Research 10, 1341–1366 (2009)

[7] Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science (New York, N.Y.) 286(5439), 531–7 (Oct 1999)

[8] Draminski, M., Rada-Iglesias, A., Enroth, S., Wadelius, C., Koronacki, J., Komorowski, J.: Monte Carlo feature selection for supervised classification. Bioinformatics 24(1), 110–117 (Nov 2008)

[9] Dudoit, S., Popper-Shaffer, J., Boldrick, J.C.: Multiple Hypothesis Testing in Microarray Experiments (2002)

[10] Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection. Journal of Machine Learning Research 3, 1157–1182 (2003)

[11] Breiman, L.: Random forests. Machine Learning 45, 5–32 (2001)

[12] L. Talavera. An evaluation of filter and wrapper methods for feature selection in categorical clustering. In In: proceeding of 6th International Symposium on Intelligent Data Analysis, pages 440–451, 2005. Madrid, Spain.